

**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»**

ФАКУЛЬТЕТ ПРИКЛАДНОЇ МАТЕМАТИКИ

Кафедра системного програмування і спеціалізованих комп'ютерних систем

«До захисту допущено»

Завідувач кафедри

_____ Віталій РОМАНКЕВИЧ

(підпис) (ініціали, прізвище)

“ ____ ” червня 20 ____ р.

Дипломний проєкт

на здобуття ступеня бакалавра

за освітньо-професійною програмою «Спеціалізовані комп'ютерні системи»

зі спеціальності **123 «Комп'ютерна інженерія»**

на тему: Редактор для створення симетричних зображень

Виконав: студент IV курсу, групи КВ-63

Самофалов Андрій Вікторович

_____ (підпис)

Керівник доц. каф. СПіСКС, к. т. н., с.н.с. Боярінова Юлія Євгенівна

(посада, науковий ступінь, вчене звання, прізвище та ініціали)

_____ (підпис)

Консультант з нормоконтролю, доц.каф.СПСКС, к.т.н. Клятченко Я.М.

_____ (підпис)

Рецензент _____

_____ (посада, науковий ступінь, вчене звання, науковий ступінь, прізвище та ініціали)

_____ (підпис)

Засвідчую, що у цьому дипломному проєкті
немає запозичень з праць інших авторів без
відповідних посилань.

Студент _____

(підпис)

Київ – 2020 року

**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»**

ФАКУЛЬТЕТ ПРИКЛАДНОЇ МАТЕМАТИКИ

Кафедра системного програмування і спеціалізованих комп'ютерних систем

Рівень вищої освіти – перший (бакалаврський)

Спеціальність 123 «Комп'ютерна інженерія»

Освітньо-професійна програма «Спеціалізовані комп'ютерні системи»

ЗАТВЕРДЖУЮ

Завідувач кафедри

_____ Віталій РОМАНКЕВИЧ

«___» _____ 2019 р.

ЗАВДАННЯ

на дипломний проєкт студента

Самофалову Андрію Вікторовичу

1. Тема проєкту «Редактор для створення симетричних зображень», керівник проєкту Боярінова Юлія Євгенівна, доц. каф. СПіСКС, к. т. н., с.н.с., затверджені наказом по університету від «25» травня 2020 р. №1181-С
2. Термін подання студентом проєкту ____ .06.2020
3. Вихідні дані до проєкту: див. Технічне завдання
4. Зміст пояснювальної записки
 - аналіз існуючих програм для створення симетричних зображень
 - огляд технологій, використаних при розробці редактору
 - розробка редактору для створення симетричних зображень
5. Перелік графічного матеріалу (із зазначенням обов'язкових креслеників, плакатів, презентацій тощо)
 - Ієрархія викликів функцій у редакторі. Схема структурна.
 - Структура елементів редактору. Схема структурна.
 - Малювання зображень з двома осями симетрії з використанням

- модифікаторів симетрії. Схема алгоритму.
- Малювання ліній великої товщини за допомогою кіл. Схема алгоритму.

6. Консультанти розділів проєкту*

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Нормконтроль	Клятченко Я.М., доцент		

7. Дата видачі завдання 15.11.2019

Календарний план

№ з/п	Назва етапів виконання дипломного проєкту	Термін виконання етапів проєкту	Примітка
1.	Вивчення літератури за тематикою проєкту	02.02.2020	
2.	Розроблення та узгодження технічного завдання	03.04.2020	
3.	Аналіз існуючих рішень	05.04.2020	
4.	Підготовка матеріалів першого розділу дипломного проєкту	14.04.2020	
5.	Підготовка матеріалів другого розділу дипломного проєкту	30.04.2020	
6.	Підготовка матеріалів третього розділу дипломного проєкту	14.05.2020	
7.	Підготовка графічної частини дипломного проєкту	15.05.2020	
8.	Оформлення документації дипломного проєкту	17.05.2020	
9.	Попередній огляд матеріалів диплому на кафедрі	20.05.2020	

Студент

(підпис)

Андрій САМОФАЛОВ

Керівник проєкту

(підпис)

Юлія БОЯРІНОВА

* Консультантом не може бути зазначено керівника дипломного проєкту.

АНОТАЦІЯ

Кваліфікаційна робота включає пояснювальну записку (52 с., 16 рис., 2 додатки).

Об'єкт розробки – редактор для створення симетричних зображень з можливістю використання модифікаторів симетрії.

Розроблений редактор дозволяє: встановлювати бажаний розмір зображення; змінювати колір та розмір пензля; малювати зображення; вибирати бажаний тип симетрії; створювати та зберігати зображення. Редактор був написаний за допомогою HTML/CSS, мови програмування JavaScript та фреймворку Bootstrap 4.

В ході розробки:

- проведено аналіз існуючих рішень;
- сформульовані вимоги до взаємодії користувача з програмою;
- розроблено архітектуру редактору;
- створено графічний інтерфейс користувача.

Цей редактор може бути легко доповнений новими функціями для покращення взаємодії користувача з програмою, стати плагіном для якогось існуючого графічного редактору, або стати основою для створення нового.

Ключові слова: РЕДАКТОР, СИМЕТРІЯ, ЗОБРАЖЕННЯ, МОВА ПРОГРАМУВАННЯ, ФРЕЙМВОРК, ПЛАГІН, HTML, CSS, JAVASCRIPT, BOOTSTRAP 4.

ABSTRACT

Qualification work includes an explanatory note (52 pp., 16 fig., 2 appendices).

The object of development is an editor for creating symmetrical images, with the possibility of using symmetry modifiers.

The developed editor allows: to set the desired image size; change the color and size of the brush; draw images; choose the desired type of symmetry; create and save images. The editor was written using HTML/CSS, JavaScript programming language and Bootstrap 4 framework.

During development:

- analysis of existing solutions;
- formulated requirements for user interaction with the program;
- the architecture of the editor is developed;
- a graphical user interface has been created.

This editor can be easily supplemented with new features to improve user interaction with the program, become a plug-in for existing graphics editors, or become the basis for creating a new one.

Keywords: EDITOR, SYMMETRY, IMAGE, PROGRAMMING LANGUAGE, FRAMEWORK, PLUGIN, HTML, CSS, JAVASCRIPT, BOOTSTRAP 4.

Поз.	Формат	ПОЗНАЧЕННЯ	НАЙМЕНУВАННЯ	Кількість аркушів	№ прим.	Примітки
	A4	ІАЛЦ.045440.004 ПЗ	Редактор для створення симетричних зображень	52		
			Пояснювальна записка			
	A4	ІАЛЦ.045440.005 Д1	Ієрархія викликів функцій у редакторі.	1		
			Схема структурна			
	A4	ІАЛЦ.045440.006 Д2	Структура елементів редактору.	1		
			Схема структурна			
	A4	ІАЛЦ.045440.007 Д3	Малювання зображень з двома осми симетрії з використанням модифікаторів симетрії.	1		
			Схема алгоритму			
	A4	ІАЛЦ.045440.008 Д4	Малювання ліній великої товщини за допомогою кіл.	1		
			Схема алгоритму			
ІАЛЦ.045440.003 ТП						
Змін.	Арк.	№ докум.	Підпис	Дата		
Розробив		Самофалов А.В.			Літ.	Аркуш
Перевірив		Боярінова Ю.Є.				Аркушів
Консульт.						
Н. контроль		Клятченко Я.М.			КПІ ім. Ігоря Сікорського, ФПМ КВ-63	
Зав. каф.		Романкевич В.О.				
Редактор для створення симетричних зображень				1	2	
Відомість технічного проекту						

[illegible]

ЗМІСТ

1. НАЙМЕНУВАННЯ ТА ГАЛУЗЬ РОЗРОБКИ.	2
2. ПІДСТАВА ДЛЯ РОЗРОБКИ.	2
3. МЕТА І ПРИЗНАЧЕННЯ РОБОТИ.	2
4. ДЖЕРЕЛА РОЗРОБКИ.	2
5. ТЕХНІЧНІ ВИМОГИ.	2
5.1. Вимоги до програмного продукту, що розробляється.	2
5.2. Вимоги до апаратного забезпечення.	3
5.3. Вимоги до програмного та апаратного забезпечення користувача. ...	3
6. ЕТАПИ РОЗРОБКИ.	4

					ІАЛЦ.045440.002 ТЗ	Арк.
						1
Змін.	Арк.	№ докум.	Підпис	Дата		

1. НАЙМЕНУВАННЯ ТА ГАЛУЗЬ РОЗРОБКИ

Назва розробки: Редактор для створення симетричних зображень.

Галузь застосування: графічні редактори, в яких необхідна підтримка функції малювання симетричних зображень.

2. ПІДСТАВА ДЛЯ РОЗРОБКИ

Підставою для розробки є завдання на виконання роботи першого (бакалаврського) рівня вищої освіти, затверджене кафедрою системного програмування і спеціалізованих комп'ютерних систем Національного технічного університету України «Київський політехнічний інститут імені Ігоря Сікорського».

3. МЕТА І ПРИЗНАЧЕННЯ РОБОТИ

Метою даного проекту є створення редактору симетричних зображень, виконаного у вигляді веб-сторінки, для розробки зображень з використанням різних осей та модифікаторів симетрії.

4. ДЖЕРЕЛА РОЗРОБКИ

Джерелом інформації є технічна та науково-технічна література, технічна документація, публікації в періодичних виданнях та електронні статті у мережі Інтернет.

5. ТЕХНІЧНІ ВИМОГИ

1.1 Вимоги до програмного продукту, що розробляється

- сумісність з сучасними веб-браузерами;
- можливість встановлювати бажаний розмір зображення;
- можливість змінювати колір пензля;
- можливість змінювати розмір пензля;

					ІАЛЦ.045440.002 ТЗ	Арк.
						2
Змін.	Арк.	№ докум.	Підпис	Дата		

- можливість малювати зображення;
- наявність вибору бажаного типу симетрії;
- використання модифікаторів симетрії;
- можливість створювати та зберігати зображення.

5.2 Вимоги до апаратного забезпечення

- оперативна пам'ять: 1 Гб;
- наявність доступу до мережі Wi-Fi (IEEE 802.11 b/g/n).

5.3 Вимоги до програмного та апаратного забезпечення користувача

- наявність доступу до мережі Інтернет;
- будь-який сучасний веб-браузер з підтримкою JavaScript.

					ІАЛЦ.045440.002 ТЗ	Арк.
						3
Змін.	Арк.	№ докум.	Підпис	Дата		

6. ЕТАПИ РОЗРОБКИ

№ з/п	Назва етапів виконання дипломного проекту	Термін виконання етапів
1.	Видача завдання на дипломне проектування	15.11.2019
2.	Вивчення літератури за тематикою роботи	02.02.2020
3.	Розроблення та узгодження технічного завдання	03.04.2020
4.	Розроблення структури редактору	07.04.2020
5.	Розроблення дизайну та графічних елементів	08.04.2020
6.	Програмна реалізація редактору	09.04.2020
7.	Тестування редактора	12.04.2020
8.	Підготовка матеріалів текстової частини проекту	14.04.2020
9.	Підготовка матеріалів графічної частини проекту	15.05.2020
10.	Оформлення технічної документації проекту	17.05.2020

					ІАЛЦ.045440.002 ТЗ	Арк.
Змін.	Арк.	№ докум.	Підпис	Дата		4

3MICT

ПЕРЕЛІК ТЕРМІНІВ, СКОРОЧЕНЬ ТА ПОЗНАЧЕНЬ.	3
ВСТУП.	4
1. АНАЛІЗ ІСНУЮЧИХ ПРОГРАМ ДЛЯ СТВОРЕННЯ СИМЕТРИЧНИХ ЗОБРАЖЕНЬ.	5
1.1. Різновиди графічних редакторів.	5
1.2. Аналіз існуючих рішень.	8
1.3. Обґрунтування теми дипломного проєкту.	13
Висновки до розділу.	14
2. ОГЛЯД ТЕХНОЛОГІЙ, ВИКОРИСТАНИХ ПРИ РОЗРОБЦІ РЕДАКТОРУ.	15
2.1. Поняття веб-сайту.	15
2.2. HTML/CSS.	19
2.3. JavaScript.	24
2.4. Фреймворк Bootstrap.	26
Висновки до розділу.	28
3. РОЗРОБКА РЕДАКТОРУ ДЛЯ СТВОРЕННЯ СИМЕТРИЧНИХ ЗОБРАЖЕНЬ.	29
3.1. Огляд структурних елементів редактору.	29
3.2. Малювання ліній.	34
3.3. Перетворення координат та малювання при кожному типі симетрії.	43
3.4. Модифікатори симетрії.	45

					ІАЛЦ.045440.004 ПЗ					
Змін	Арк.	№ докум.	Підпис	Дата						
Розробив		Самофалов А.В.			<i>Редактор для створення симетричних зображень</i> <i>Пояснювальна записка</i>			Літ.	Аркуш	Аркушів
Перевірив		Боярінова Ю.Є.							1	52
Н. контроль		Клятченко Я.М.						КПІ ім. Ігоря Сікорського, ФПМ КВ-63		
Затвердив		Романкевич В.О.								

Висновки до розділу.	47
4. ВИСНОВКИ.	48
СПИСОК ВИКОРИСТАНИХ ЛІТЕРАТУРНИХ ДЖЕРЕЛ.	50

ДОДАТКИ

Додаток 1. Копії графічного матеріалу.

ІАЛЦ.045440.005 Д1. Ієрархія викликів функцій у редакторі. Схема структурна.

ІАЛЦ.045440.006 Д2. Структура елементів редактору. Схема структурна.

ІАЛЦ.045440.007 Д3. Малювання зображень з двома осями симетрії з використанням модифікаторів симетрії. Схема алгоритму.

ІАЛЦ.045440.008 Д4. Малювання ліній великої товщини за допомогою кіл. Схема алгоритму.

Додаток 2. Фрагменти програмного коду.

					ІАЛЦ.045440.004 ПЗ	Арк.
Зм	Лист	№ докум.	Підп.	Дата		2

ПЕРЕЛІК ТЕРМІНІВ, СКОРОЧЕНЬ ТА ПОЗНАЧЕНЬ

CSS – Cascading Style Sheets – каскадні таблиці стилів;

HTML – HyperText Markup Language – мова розмітки гіпертекстових документів;

Photoshop CC – Photoshop Creative Cloud;

Q&A – questions and answers – запитання й відповіді;

URL – Uniform Resource Locator – стандартизована адреса ресурсу в мережі Інтернет;

W3C – World Wide Web Consortium – Всесвітній веб-консорціум;

WWW – World Wide Web – всесвітня мережа.

					ІАЛЦ.045440.004 ПЗ	Арк.
						3
Зм	Лист	№ докум.	Підп.	Дата		

ВСТУП

На сьогоднішній день великим попитом користуються унікальні зображення, створені за допомогою графічних редакторів. Ці зображення дозволяють розробникам нових програмних продуктів краще передавати ідеї, що закладені в їх програмах, таким чином покращуючи враження користувачів від цих програм. У наш час існує багато графічних редакторів, які дозволяють створювати симетричні й не тільки зображення. Але часто розробникам необхідна лише деяка функція, як-от малювання симетричних зображень, а існуючі редактори можуть бути дуже складними до роботи для тих, хто з ними ще не працював.

Завданням на дипломний проєкт є розробка редактору для створення симетричних зображень. Основною відмінністю створюваного редактору від вже існуючих рішень буде можливість відключати відображення на будь-яку частину від осі симетрії. Завдяки цьому користувач буде мати змогу з легкістю розробляти зображення, які не повністю симетричні, не змінюючи при цьому вибраної симетрії.

					ІАЛЦ.045440.004 ПЗ	Арк.
						4
Зм	Лист	№ докум.	Підп.	Дата		

1. АНАЛІЗ ІСНУЮЧИХ ПРОГРАМ ДЛЯ СТВОРЕННЯ СИМЕТРИЧНИХ ЗОБРАЖЕНЬ

1.1. Різновиди графічних редакторів

Графічний редактор – це програма, що дозволяє створювати та обробляти зображення за допомогою комп'ютера: малювати лінії, редагувати зображення, розфарбовувати області екрану, створювати написи й фігури та інше. Графічний редактор зберігає зображення в графічних файлах, а також дозволяє виводити зображення на принтер. Деякі з графічних редакторів забезпечують можливість створення зображень тривимірних об'єктів, симулювати їх перетини та розвороти [1].

Графічні редактори поділяють на два типи [2]:

- Растровий графічний редактор;
- Векторний графічний редактор.

Растровий графічний редактор – це спеціалізована комп'ютерна програма, яка дозволяє користувачам інтерактивно створювати та редагувати растрові зображення на екрані комп'ютера та зберігати їх в одному з багатьох "растрових" форматів, таких як JPEG, PNG, GIF.

Растрові зображення складаються з рядків і стовпців спеціальних крапок, які утворюють матрицю. Ці крапки називаються пікселями, і, як правило, отримані зображення є більш фотореалістичними, тому растрові зображення використовуються в цифровій фотографії.

Піксель – це мінімальна ділянка зображення. Кожен піксель має свої розташування і колір, які ніяким чином не залежать від відповідних значень інших пікселів.

Якість растрового зображення залежить від кількості пікселів у цьому зображенні. Кожен піксель займає деяку пам'ять на комп'ютері, а загальний об'єм пам'яті, який займає зображення, можна знайти за наступною формулою:

$V = K * i$, де V – об'єм пам'яті, який зайнятий зображенням;

					ІАЛЦ.045440.004 ПЗ	Арк.
						5
Зм	Лист	№ докум.	Підп.	Дата		

К – кількість пікселів;

і – глибина кольору.

Список деяких растрових графічних редакторів:

Комерційні:

- ACDSee Photo Editor;
- Adobe Photoshop;
- Adobe Image Editor;
- Autodesk SketchBook Pro;
- Corel Painter.

Безкоштовні:

- GIMP;
- GNU Paint;
- MyPaint;
- Pinta.

Векторний графічний редактор - це спеціалізована комп'ютерна програма, яка дозволяє користувачам створювати та обробляти зображення векторної графіки на комп'ютері та зберігати їх в одному з багатьох популярних форматів векторної графіки, таких як EPS, PDF, SVG, VML, або WMF.

Векторні зображення складаються з геометричних об'єктів, таких як точки, лінії, сплайни і багатокутники. Ці об'єкти зберігаються в пам'яті комп'ютера за допомогою математичних формул. Векторні графічні зображення є засобом зберігання високоточних графічних об'єктів (схеми, креслення та ін.), для яких має значення збереження високої якості зображення та чітких контурів.

Векторні редактори, як правило, краще підходять для графічного дизайну, створення макетів сторінок, типографії, логотипів, художніх ілюстрацій, наприклад, мультфільмів, графічних зображень, складних геометричних візерунків, технічних ілюстрацій, діаграм та блок-схем.

Деякі векторні редактори підтримують анімацію, а інші (наприклад, Adobe

					ІАЛЦ.045440.004 ПЗ	Арк.
						6
Зм	Лист	№ докум.	Підп.	Дата		

Flash, Animatron, або Synfig Studio) були спеціально розроблені для створення анімованої графіки. Як правило, векторна графіка більше підходить для анімації, хоча й є інструменти на основі растрової графіки.

Векторні редактори тісно пов'язані з програмним забезпеченням для роботи з публікаціями, такі як Adobe InDesign або Scribus, що також зазвичай включає деякі інструменти векторного малювання (як правило, менш потужні, ніж ті, що використовуються в автономних векторних редакторах).

Для систем автоматизованого проектування також існують спеціальні векторні редактори. Вони не підходять для художньої чи декоративної графіки, але мають у своєму арсеналі багато інструментів та бібліотек об'єктів, які використовуються для забезпечення точності та відповідності стандартам малюнків та креслень.

Нарешті, існують редактори для комп'ютерної 3D графіки, такі як Maya, Blender, або 3D Studio Max. Їх можна розглядати як розширення традиційних 2D векторних редакторів, оскільки вони поділяють деякі загальні поняття та інструменти.

Список деяких векторних графічних редакторів:

Комерційні:

- Adobe Illustrator;
- CorelDRAW;
- Microsoft Visio;
- Sketch.

Безкоштовні:

- Apache OpenOffice Draw;
- Inkscape;
- Synfig.

Редактори векторної графіки часто контрастують з растровими графічними редакторами, проте їх можливості доповнюють одні інших. Технічна різниця між векторними та растровими редакторами впливає з різниці між

					ІАЛЦ.045440.004 ПЗ	Арк.
						7
Зм	Лист	№ докум.	Підп.	Дата		

векторними та растровими зображеннями. Векторна графіка створюється математично за допомогою геометричних формул. Кожен елемент створюється та обробляється чисельно; по суті, використовуючи декартові координати для розміщення ключових точок, а потім математичний алгоритм з'єднує крапки та визначає кольори.

Розширені растрові редактори, такі як GIMP та Adobe Photoshop, використовують векторні методи для загального компоновання та елементів, таких як текст, але також оснащені інструментами для обробки растрових зображень до пікселя і часто мають спеціальні можливості для цього, такі як яскравість / контраст і, навіть, додавання «освітлення» до зображення чи фотографії.

1.2. Аналіз існуючих рішень

Adobe Photoshop

Adobe Photoshop – це растровий графічний редактор, розроблений та опублікований Adobe Inc. для Windows та macOS. Спочатку він був створений у 1988 році Томасом та Джоном Ноллом [3].

Photoshop використовує систему редагування на основі шарів, яка дозволяє створювати зображення та змінювати їх з використанням декількох накладок, що підтримують прозорість. Шари також можуть діяти як маски або фільтри, змінюючи основні кольори. Цим шарам також можуть додаватися тіні та інші необхідні ефекти. Також є підтримка масок, альфа-композицій та кілька кольорових моделей, включаючи RGB, CMYK, CIELAB, точковий колір та дуотон. Photoshop використовує власні формати файлів PSD та PSB для підтримки цих функцій. Окрім растрової графіки, це програмне забезпечення також має обмежені можливості редагування чи візуалізації тексту та векторної графіки, а також 3D графіки та відео. Набір функцій можна розширити плагінами.

Плагіни Photoshop (або плагіни) – це додаткові програми, спрямовані на

					ІАЛЦ.045440.004 ПЗ	Арк.
						8
Зм	Лист	№ докум.	Підп.	Дата		

надання додаткових ефектів зображення, або виконання завдань, які неможливо, або важко виконати, використовуючи лише Adobe Photoshop. Плагіни можна відкрити з програми Photoshop та декількох інших програм редагування зображень (сумісних з відповідними специфікаціями Adobe).

Існують різні типи плагінів, такі як фільтр, експорт, імпорт, вибір, корекція кольорів та автоматизація. Найпопулярніші плагіни - це додатки з фільтрами (також відомі як плагіни 8bf), доступні в меню Фільтр у Photoshop. Фільтри плагінів можуть або змінювати поточне зображення, або створювати вміст. Нижче наведено кілька популярних типів плагінів:

- Плагіни для корекції кольорів;
- Плагіни для спеціальних ефектів;
- Плагіни для 3D-ефектів;
- Adobe Camera Raw (також відомий як ACR і Camera Raw) – це спеціальний плагін, що постачається безкоштовно компанією Adobe, який використовується в основному для читання та обробки файлів із «сирої» картинки, щоб отримані зображення могли бути оброблені Photoshop.

Файл «сирих» зображень камери містить мінімально оброблені дані з датчика зображення цифрової камери, сканера кінофільму, або іншого сканера зображення. Сирі файли названі так, тому що вони ще не оброблені і тому не готові до друку чи редагування за допомогою графічного редактора растрових зображень.

Інтерфейс Photoshop включає функції автоматизації, для того щоб зменшити потребу в повторюваних завданнях. Опція, відома як Photoshop CC, дозволяє користувачам працювати над вмістом їх проектів з будь-якого комп'ютера.

Photoshop настільки довго був стандартом у галузі програм маніпулювання зображеннями, що його назва стала дієсловом: загальноприйнятою є думка, що зображення було "фотошопним". "Фотошопне" в цьому контексті є синонімом редагування, маніпулювання, або підробки зображень - часто незалежно від

					ІАЛЦ.045440.004 ПЗ	Арк.
						9
Зм	Лист	№ докум.	Підп.	Дата		

програмного забезпечення, яке фактично було для цього використане [4].

Photoshop використовується фотографами, графічними дизайнерами, художниками відеоігор, рекламодавцями та усіма, кому необхідно працювати з редагуванням зображень. Photoshop CC сумісний з комп'ютерами Mac на базі Intel та комп'ютерами з ОС Windows.

Функцією Photoshop, що нас цікавить, є Paint Symmetry, тому розглянемо її детальніше [5].

Інструмент Paint Symmetry допомагає малювати дзеркальні зображення в будь-якій площині. Це дозволяє використовувати один, або кілька шляхів для визначення межі симетрії - все, що ви малюєте на одній стороні, з'являється з іншого боку, ніби зображення було віддзеркалене. Як можна побачити з назви, ця функція працює з інструментами:

- Paintbrush;
- Mixer Brush;
- Pencil;
- Eraser.

Піктограма метелика на панелі інструментів повідомляє про те, що функція «Paint Symmetry» активна. Існує декілька доступних типів симетрії – вертикальна, горизонтальна, подвійну вісь, діагональна, хвиляста, коло, спіральна, паралельні лінії, радіальна та мандала.

- Для радіальної симетрії можна вказати будь-яку кількість сегментів від 2 (мінімум) до 12 (максимум).

Під час малювання за допомогою радіальної симетрії Photoshop повторює один штрих пензля навколо центральної точки, або радіальної осі. Наприклад, встановлення кількості сегментів для радіальної симетрії до шести призводить до того, що один штрих повторюється шість разів навколо центральної точки.

- Для мандала-симетрії можна вказати будь-яку кількість сегментів від 3 (мінімум) до 10 (максимум).

					ІАЛЦ.045440.004 ПЗ	Арк.
						10
Зм	Лист	№ докум.	Підп.	Дата		

Під час малювання за допомогою мандала-симетрії, Photoshop спочатку дзеркально відображає, а потім повторює один штрих пензля навколо центральної точки або радіальної осі. Наприклад, встановлення кількості сегментів для мандала-симетрії до шести призводить до відображення одного штриху пензля та потім повторення цього штриху шість разів навколо центральної точки.

Цікавою функцією є також створення декількох шляхів симетрії. Можна скопіювати / перетворити / вставити контури симетрії для створення декількох одночасних результатів. Найкраще це проглядається при роботі з дуже маленькими пензликами, приблизно по 4 пікселя, або менше. Наприклад, якщо вибрати вертикальний шлях симетрії, скопіювати та вставити його чотири, або п'ять разів, то при малюванні результат буде продубльовано для кожного створеного шляху.

Можна створювати та зберігати безліч різних контурів симетрії в заданому документі, що дозволяє продовжувати роботу після закриття та повторного відкриття документа, або при використанні спільного доступу до документів. Існують також деякі обмеження, такі як обмеження на використання лише звичайних пензлів - відсутні живі щітки, такі як міксер або ерозійні наконечники. Це також означає відсутність плям, градієнтів, виділень чи додаткових шляхів; які, однак, можуть бути стертими за допомогою симетрії.

Також треба зауважити, що прості шляхи будуть працювати найкраще. Шляхи з використанням складних кривих дуже швидко перейдуть у хаос, а використання занадто багато дублюючих контурів з великими пензлями може серйозно розігріти процесор комп'ютера.

Autodesk 3ds Max

Autodesk 3ds Max, раніше 3D Studio і 3D Studio Max – це професійна програма для 3D-графіки та створення 3D-анімацій, моделей, ігор, зображень [6]. Вона розробляється компанією Autodesk Media and Entertainment, та ж компанія, що відповідає за такі програми, як Maya та AutoCAD. Autodesk 3ds Max має

					ІАЛЦ.045440.004 ПЗ	Арк.
						11
Зм	Лист	№ докум.	Підп.	Дата		

можливості моделювання та гнучку архітектуру плагінів і повинна використовуватися на платформі Microsoft Windows. Цю програму часто використовують розробники відеоігор, архітектурні візуалізаційні студії, багато телевізійних комерційних студій та інші; для створення кіноефектів та попередньої візуалізації фільму. Із інструментів для моделювання та створення анімації є шейдери (такі, як оклюзія навколишнього середовища та розсіювання під поверхнею), динамічне моделювання, глобальне освітлення, системи частинок, радіостійкість, звичайне створення карт та візуалізація.

3ds Max часто використовується для моделювання персонажів та анімації, а також для надання фотореалістичних зображень будівель та інших об'єктів. Що стосується моделювання, то 3ds Max не має собі рівних у швидкості та простоті.

Програмне забезпечення може обробляти кілька етапів конвеєра анімації, включаючи попередню візуалізацію, макет, камери, моделювання, текстурування, такелаж, анімацію, VFX, освітлення та візуалізацію.

Функцією Autodesk 3ds Max, для створення симетрії є Symmetry Modifier [7]. Цей модифікатор унікальний тим, що дозволяє виконувати три поширені завдання моделювання:

- Віддзеркалити сітку навколо площин X, Y або Z;
- Зріз сітки, при необхідності видаляючи деталі;
- Автоматично з'єднує вершини вздовж загального шва.

Модифікатор Symmetry може бути застосований до будь-якої геометрії, або для анімування дзеркального, або нарізаючого ефекту.

CorelDRAW

CorelDRAW – це редактор векторної графіки, розроблений корпорацією Corel [8]. Це також назва графічного набору Corel, який включає редактор растрових зображень Corel Photo-Paint, а також інші програми, пов'язані з графікою. Програмне забезпечення є надійним графічним набором, що забезпечує багато функцій для редагування графіки. Ці функції включають регулювання контрасту, балансування кольорів, додавання до зображень

					ІАЛЦ.045440.004 ПЗ	Арк.
						12
Зм	Лист	№ докум.	Підп.	Дата		

спеціальних ефектів, таких як рамки, і вони можуть працювати з декількома шарами та кількома сторінками. Програма CorelDRAW призначена для редагування двовимірних зображень, таких як логотипи та плакати.

Функцією CorelDRAW, для створення симетрії є Symmetry [9], яка взагалом не дуже сильно відрізняється від вже розглянутої PaintSymmetry в Adobe Photoshop.

1.3. Обґрунтування теми дипломного проєкту

Неважко побачити, що майже всі об'єкти, що нас оточують, мають у собі ту чи іншу симетрію. Тому можливість створення симетричних зображень за допомогою комп'ютера є дуже цінною. Наприклад, симетричний модифікатор особливо корисний при моделюванні персонажів, предметів, креслень, будівництві кораблів, або літальних апаратів.

Розроблюваний редактор як раз і розрахований на те, щоб дати користувачам, навіть без досвіду роботи з графічними редакторами, можливість створювати симетричні зображення. Особливістю програми буде мінімалістичний дизайн, для того щоб не відволікати користувачів від роботи над їх проєктами. А ключовою відмінністю створюваного редактору від вже існуючих рішень буде можливість відключати відображення на будь-яку частину від осі симетрії. Також користувачі, за бажанням, можуть взагалі відключати осі симетрії, щоб продовжити малювання в звичайному режимі.

Редактор передбачає наступні функції:

- встановлювати бажаний розмір зображення;
- змінювати колір та розмір пензля;
- вибирати бажаний тип симетрії;
- створювати та зберігати зображення.

Сама програма буде у вигляді веб-сторінки, так як майже у всіх користувачів є доступ до мережі Інтернет, і не всі користувачі можуть мати змогу чи бажання для встановлення якихось додаткових програм на їх комп'ютери.

					ІАЛЦ.045440.004 ПЗ	Арк.
Зм	Лист	№ докум.	Підп.	Дата		13

При розробці редактору будуть використані HTML/CSS, мова програмування Javascript та фреймворк Bootstrap 4, які будуть докладніше розглянуті в наступному розділі.

Висновки до розділу

В даному розділі було розглянуто основні графічні редактори, які використовуються для створення симетричних зображень. Вони всі є успішними комерційними продуктами та лідерами в своїй сфері застосування. А їх програмні інтерфейси надають користувачам потужні інструменти для обробки будь-яких зображень. Темою на дипломний проєкт була обрана розробка редактору для створення симетричних зображень, тому що при всьому різноманітті функцій, в вище зазначених редакторах не має можливості відключати відображення на будь-яку частину від заданих осей симетрії. Через це користувачам доводиться відключати взагалі дзеркальний режим, щоб домалювати якусь несиметричну деталь. Проте наявність цієї опції дозволила б уникнути цих незручностей та могла сприяти кращому користувацькому досвіду.

					ІАЛЦ.045440.004 ПЗ	Арк.
						14
Зм	Лист	№ докум.	Підп.	Дата		

2. ОГЛЯД ТЕХНОЛОГІЙ, ВИКОРИСТАНИХ ПРИ РОЗРОБЦІ РЕДАКТОРУ

2.1. Поняття веб-сайту

Сайт або веб-сайт є центральним розташуванням веб-сторінок, які пов'язані між собою та до яких можна дістатись відвідавши домашню сторінку веб-сайту за допомогою браузера [10].

Веб-сторінка – це документ, зазвичай написаний у HTML, який переглядається в Інтернет-браузері. Для доступ до веб-сторінки необхідно ввести URL-адресу в адресний рядок браузера. Веб-сторінка може містити текст, графіку та гіперпосилання на інші веб-сторінки та файли.

Веб-сторінка часто використовується для надання інформації користувачам, включаючи картинки чи відео, які допомагають проілюструвати важливі теми. Веб-сторінка може також використовуватися як метод продажу товарів чи послуг.

Типи веб-сайтів

Сьогодні в Інтернеті є мільярди веб-сайтів, які можна розділити на наступні типи категорій веб-сайтів. Також треба мати на увазі, що веб-сайт можна поділити на більш ніж одну з наступних категорій. Наприклад, веб-сайт також може бути форумом, веб-поштою, блогом, або пошуковою системою.

- Архівний веб-сайт – це веб-сайт, який веде облік вмісту одного або декількох інших веб-сайтів. Інтернет-архів – найкращий приклад веб-сайту архіву.
- Блог (веб-журнал) – це веб-сайт, який часто створюється особою, щоб вести список записів, які їх цікавлять. Веб-сайт мікроблогів – це ще одна популярна форма веб-сайту для ведення блогів, що обмежує кількість користувачів, які можуть публікувати записи в кожному розділі блогу. Twitter є прикладом мікроблогу в соціальній мережі.
- Бізнес-сайт, або корпоративний веб-сайт створений для надання інформації про акаунти клієнтів, партнерів, клієнтів та потенційних

клієнтів, а також надання доступу до них.

- Веб-сайт спільноти – це веб-сайт, або розділ веб-сайту, який допомагає зблизити користувачів, які відвідують цей сайт, використовуючи чат, форуми або іншу форму оголошень.
- Веб-сайт знайомств – це сайт, створений для того, щоб допомогти зв'язати людей, які можуть бути зацікавлені у зустрічі з іншими людьми чи знайомстві з іншими людьми. Більшість веб-сайтів для знайомств вимагають невеликої плати, опису себе та часто мають ряд питань, щоб допомогти знайти людей, які найкраще відповідають їх інтересам.
- Веб-сайт з оглядом – це будь-який веб-сайт, який фокусується на відгуках про товар чи послугу. Оглядові сайти можуть переглядати різні речі, такі як фільми чи продукти. Крім того, на багатьох сайтах електронної комерції є відгуки людей, які придбали товар, який вони продають.
- Веб-сайт пошукової системи – це веб-сайт, присвячений допомозі людям знаходити інформацію в Інтернеті. Google є прикладом веб-сайту пошукової системи.
- Веб-сайт соціальних мереж з'єднує користувачів із друзями, родиною, знаменитостями, групами та організаціями. Послуга зазвичай безкоштовна за умови, що веб-сайт може збирати та використовувати інформацію стосовно користувачів. Facebook та Twitter є прикладами веб-сайту соціальних мереж.
- Веб-сайт довідки, веб-сайт із запитаннями та відповідями (також англ. – Q&A website) – це веб-сайт, на якому кожен може публікувати запитання, а інші користувачі можуть допомогти відповісти на ці запитання.
- Веб-сайт для спільного використання медіа - це будь-який веб-сайт, який спеціалізується на тому, щоб відвідувачі могли ділитися одним

або кількома видами медіа. Наприклад, YouTube - це сайт для обміну відеомедіа. SoundCloud - це сайт для обміну музикою. Flickr - це місце для обміну фотографіями. DeviantArt - це сторінка для спільного використання мистецтва.

- Веб-сайт соціальних новин – це веб-сайт, який генерує свій вміст за рахунок своїх користувачів, і, опублікувавши його, всі інші користувачі можуть проголосувати, якщо їм сподобалась певна історія. Reddit є прикладом веб-сайту соціальних новин, який дозволяє кожному публікувати майже все. Це прекрасне місце для пошуку речей, які найбільше зацікавили б вас в Інтернеті.
- Веб-сайт пошти – це веб-сайт, який людина може переглядати, надсилати та отримувати електронні листи без необхідності встановлювати спеціалізоване програмне забезпечення. Прикладом програми веб-пошти є Gmail.
- Вікі веб-сайт – це сайт, створений за допомогою програмного забезпечення Wiki, і його часто редагують та оновлюють більше однієї людини. Він слугує своєрідною енциклопедією за якоюсь певною тематикою.
- Змістовий веб-сайт та інформаційний веб-сайт створені з метою відображення унікального контенту, який часто пов'язаний із певною категорією. Інші категорії можуть включати політичний веб-сайт із вмістом, що стосується політики чи політичного погляду, або релігійний веб-сайт із інформацією про конкретну релігію.
- Веб-сайт електронної комерції (електронна комерція) – це будь-який веб-сайт, створений з метою продажу онлайн товарів чи послуг. Amazon є типовим прикладом веб-сайту електронної комерції. Веб-сайт електронної комерції може бути ще більше розбитий на одну з наступних підкатегорій.
 - Партнерський веб-сайт – це веб-сайт, створений з метою продажу

товарів сторонніх виробників. Наприклад, Amazon має партнерську програму для будь-кого, хто має посилання на свій сайт та здійснить комісію при придбанні продукції. Партнерський веб-сайт не слід плутати з веб-сайтом електронної комерції.

- Веб-сайт аукціону – це веб-сайт, який дозволяє іншим продавати свої товари чи послуги. Наприклад, eBay – один з найвідоміших веб-сайтів онлайн-аукціонів.
- Веб-сайт із класифікованими оголошеннями – це веб-сайт, який дозволяє кожному перелічити товари чи послуги, як правило, безкоштовно, або за невелику ціну.
- Веб-сайт для краудфандингу створюється для підтримки бізнесу, особи чи іншої справи шляхом одноразового, або щомісячного платежу. Прикладом веб-сайту для краудфандингу є Kickstarter.
- Ігровий веб-сайт – це будь-який веб-сайт, який містить ігри, в які можна грати на цьому веб-сайті. Часто ці онлайн-ігри створюються за допомогою HTML5, Flash або Java. Ігрові веб-сайти не слід плутати з веб-сайтом із вмістом ігор, на якому є вміст, релевантний для ігор, і на ньому немає фактичних ігор.
- Урядовий веб-сайт – це районний, місцевий чи державний веб-сайт, створений для інформування громадськості про державний бізнес та послуги. Також може бути створений веб-сайт місцевого самоврядування для сприяння розвитку туризму.
- Дзеркальний веб-сайт – це повний дублікат іншого веб-сайту, який використовується, коли веб-сайт перевантажений. Це допомагає збільшити швидкість веб-сайту при його використанні в різних куточках світу.
- Веб-сайт новин – це сайт, присвячений наданню останніх місцевих чи світових новин. Сайт новин також може бути присвячений певній темі. Наприклад, багато веб-сайтів із новинами про комп'ютери присвячені

розмові про комп'ютери та сучасні технології.

- Персональний веб-сайт – це сайт, створений особою, яка розповідає про своє особисте життя, сім'ю, життєвий досвід і, можливо, містить резюме. Сьогодні багато людей створюють особисті веб-сайти як блог, або використовують веб-сайт у соціальних мережах як місце для зберігання інформації про себе.
- Веб-сайт особистості – це будь-який веб-сайт, який стосується будь-якої відомої людини, наприклад, художника, музиканта, футболіста, автора відомої книги чи будь-якої іншої людини. Ці типи веб-сайтів створюються особами, якимось пов'язаними з цією відомою людиною, наприклад, публіцистом, агентством чи фанатом.

2.2. HTML/CSS

HTML

HTML означає Hypertext Markup Language. Гіпертекст – це текст, який міститься у посиланнях на інші тексти, які користувачі можуть негайно переглянути. Також це метод, за допомогою якого можна пересуватися в Інтернеті – натискаючи на спеціальні елементи під назвою гіперпосилання, які переводять на наступну сторінку. Слово «гіпер» просто означає, що цей процес не є лінійним, тобто можна перейти в будь-яке місце в Інтернеті, коли завгодно, натискаючи на посилання – не існує чіткої інструкції, у якому порядку вони мають відкриватись. Розмітка – це те, що теги HTML роблять для певних елементів сторінки. Наприклад, вони можуть бути використані для зміни типу тексту (зробити з нього курсив, жирне накреслення, тощо).

HTML – це мова, оскільки вона має власні кодові слова та синтаксис, як і будь-яка інша мова. Вона дозволяє користувачеві створювати та структурувати розділи, абзаци, заголовки, посилання та цитування для веб-сторінок та додатків. HTML не є мовою програмування, тобто вона не має можливості створювати динамічну функціональність. Натомість, це дозволяє організувати та

					ІАЛЦ.045440.004 ПЗ	Арк. 19
Зм	Лист	№ докум.	Підп.	Дата		

форматувати документи, подібно до текстових редакторів.

HTML винайшов Тім Бернерс-Лі, фізик з науково-дослідного інституту CERN у Швейцарії. Він запропонував ідею створення Інтернет-системи гіпертексту [11].

Перша версія HTML була опублікована у 1991 році, та складалася з 18 тегів HTML. Відтоді кожна нова версія мови HTML містила нові теги та атрибути (модифікатори тегів) для розмітки. Теги (їх також називають елементами) можна віднести до складових частин веб-сторінок. Вони створюють ієрархію, яка структурує вміст у розділи, параграфи, заголовки та інші блоки. На даний момент існує 140 тегів HTML, хоча деякі з них вже застаріли (не підтримуються сучасними браузером).

Через швидке зростання популярності, HTML зараз вважається офіційним веб-стандартом. Характеристики HTML підтримуються та розробляються Всесвітнім консорціумом веб-сторінок (W3C). Можна будь-коли перевірити останню версію на веб-сайті W3C [12].

Плюси та мінуси HTML

Як і більшість речей, HTML оснащена кількома сильними сторонами та обмеженнями.

Плюси:

- Широко використовувана мова з великою кількістю ресурсів і величезною спільнотою;
- Запускається у кожному веб-браузері;
- Має пласку криву навчання;
- Має відкритий код та абсолютно безкоштовна;
- Чиста і послідовна розмітка;
- Офіційні веб-стандарти підтримуються Всесвітнім консорціумом веб-сторінок (W3C);
- Легко інтегрується з такими мовами, як PHP та Node.js.

Мінуси:

- В основному використовується для статичних веб-сторінок. Для динамічної функціональності може знадобитися використання JavaScript, або іншої мови, наприклад PHP;
- Це не дозволяє користувачеві реалізовувати логіку. Як результат, усі веб-сторінки потрібно створювати окремо, навіть якщо вони використовують однакові елементи, наприклад колонтитули;
- Деякі браузери повільно виконують нові функції;
- Поведінку браузера іноді важко передбачити (наприклад, старіші веб-браузер не завжди мають підтримку нових тегів).

Як пов'язані HTML, CSS та JavaScript?

Хоча HTML – це потужна мова, але цього недостатньо для створення професійного і повністю динамічного веб-сайту. Її можна використовувати лише для додавання текстових елементів та створення структури вмісту.

Однак HTML надзвичайно добре працює з двома іншими мовами інтерфейсу: CSS (каскадні таблиці стилів) та JavaScript. Разом вони можуть досягти гарного користувацького враження від роботи з веб-сайтом та реалізувати розширені функції.

CSS відповідає за такі стилі, як фон, кольори, макети, інтервали та анімації.

JavaScript дозволяє додавати динамічні функції, такі як повзунки, спливаючі вікна та фотогалереї.

Якщо ми порівнюємо програмування з будівництвом будинку, то HTML визначає структуру будинку, а саме місце стін та дверей, тоді як CSS є інтер'єром, наприклад, килимами та шпалерами, які прикрашають будинок та роблять його гарним. JavaScript додає інтерактивності та означає, що ви можете відкривати двері та вмикати світло.

CSS

CSS означає Cascading Style Sheets, тобто каскадні таблиці стилів. Ця мова використовується для зміни зовнішнього вигляду розмітки веб-сторінок

					ІАЛЦ.045440.004 ПЗ	Арк.
						21
Зм	Лист	№ докум.	Підп.	Дата		

[13]. Наприклад, для визначення шрифту, розміру шрифту, ваги шрифту, його положення та інших візуальних налаштувань елементів на сторінці.

CSS допомагає веб-розробникам створювати однаковий стиль на декількох сторінках веб-сайту. Замість визначення стилю кожної таблиці та кожного блоку тексту в HTML-коді певної сторінки, загальновживані стилі потрібно визначати лише один раз у документі CSS. Після того, як стиль визначений у .css-файлі стилі, його може використовувати будь-яка сторінка, на яку посилається цей файл CSS. Крім того, CSS дозволяє легко змінювати стилі на декількох сторінках одночасно. Наприклад, веб-розробник може захотіти збільшити розмір тексту за замовчуванням з 10pt до 12pt для п'ятдесяти сторінок веб-сайту. Якщо всі сторінки посилаються на один і той же файл стилів, розмір тексту потрібно змінити лише у цьому файлі, а результат відобразиться на всіх сторінках – у цьому випадку збільшиться розмір тексту.

Хоча CSS чудово підходить для створення стилів тексту, він також корисний для форматування інших аспектів макета веб-сторінок. Наприклад, CSS може бути використаний для визначення відступів у таблицях, стилю, товщини та кольору межі таблиці та відступів навколо зображень чи інших об'єктів. CSS надає веб-розробникам більш точний контроль над тим, як виглядатимуть веб-сторінки, ніж HTML. Саме тому сьогодні більшість веб-сторінок використовують каскадні таблиці стилів [14].

Переваги CSS:

- CSS економить час – можна написати CSS один раз, а потім повторно використовувати той же файл на декількох HTML-сторінках. Ви можете визначити стиль для кожного елемента HTML і застосувати його до якомога більше веб-сторінок;
- Сторінки завантажуються швидше – при використанні CSS, не потрібно щоразу писати атрибути тегів HTML. Просто треба написати одне правило CSS тегу та застосовувати його до всіх випадків використання цього тегу. Так чином, менше коду означає

швидше завантаження.

- Покращені стилі порівняно з HTML – CSS має набагато ширший набір атрибутів, ніж HTML, тому можна створити набагато кращий вигляд сторінки HTML, порівняно з користуванням тільки атрибутами HTML.
- Сумісність декількох пристроїв – файли стилів дозволяють оптимізувати вміст для більш ніж одного типу пристроїв. Використовуючи один і той же документ HTML, різні версії веб-сайту можуть бути по-різному представлені для портативних пристроїв таких як планшети та мобільні телефони, або для друку.
- Глобальні веб-стандарти – на даний момент атрибути HTML застарівають, тому рекомендується використовувати CSS. Тож використання CSS на всіх HTML-сторінках, дозволить зробити їх сумісними з майбутніми браузером.

CSS розробляється групою в рамках W3C, яка називається Робочою групою CSS [15]. Ця група складається з представників постачальників браузерів та інших компаній, які мають інтерес до CSS. Є й інші люди, відомі як запрошені експерти, які виступають незалежними голосами; вони не пов'язані з організацією-членом.

Робоча група CSS розробляє, або уточнює нові функції CSS. Іноді, тому що конкретний браузер зацікавлений у певних можливостях, інший раз через те, що веб-дизайнери та розробники просять додати якусь певну функцію, а колись, тому що сама Робоча група визначила вимогу. CSS постійно розвивається, все більше нових функцій стає доступними. Однак ключовим у CSS є те, що всі працюють дуже наполегливо над тим, щоб ніколи не змінювати речі таким чином, щоб зламати старі веб-сайти. Веб-сайт, створений у 2000 році, використовуючи доступний тоді обмежений CSS, повинен і сьогодні відображатись у веб-браузері.

2.3. JavaScript

JavaScript – це одна з найпопулярніших мов програмування у світі з більш ніж двадцятирічною історією. Це також одна з трьох основних мов для веб-розробників.

JavaScript був створений протягом десяти днів Бранданом Ейхом, співробітником Netscape, у вересні 1995 року [16]. Спочатку мова називалася Mocha, потім ім'я мови було змінено на Mona, а потім на LiveScript, перш ніж нарешті стало називатися JavaScript. Початкові версії мови були обмежені виключно Netscape і пропонували лише обмежену функціональність, але з часом вона продовжувала зростати завдяки спільноті розробників, які працювали з нею.

У 1996 році JavaScript офіційно отримав назву ECMAScript, ECMAScript 2 з'явився в 1998 році, а ECMAScript 3 з'явився в 1999 році. Це перетворилося на сьогоднішній JavaScript, який зараз працює не тільки в різних браузерах, але і на різних пристроях, включаючи мобільні та настільні комп'ютери комп'ютери. JavaScript лише за двадцять років він перейшов від примітивної мови програмування до одного з найважливіших інструментів інструментарію веб-розробників [17].

JavaScript може бути використаний для наступних речей [18]:

- Додавання інтерактивності на веб-сайти – щоб веб-сайт був більше, ніж статична сторінка тексту;
- Розробка мобільних додатків – JavaScript призначений не лише для веб-сайтів, він використовується для створення додатків на телефони та планшети;
- Створення ігор на основі веб-браузера;
- Розробка серверної частини веб-сайтів – зазвичай JavaScript частіше використовується для написання фронтенду, але це досить універсальна мова сценаріїв, яка також може бути використана і при розробці серверної інфраструктури.

JavaScript має ряд переваг, які роблять його кращим вибором, ніж його конкуренти, особливо в певних випадках використання. Лише деякі переваги використання JavaScript включають [19]:

- Вам не потрібен компілятор, оскільки веб-браузери інтерпретують його за допомогою HTML;
- Можливість перевірити введенні дані користувача перед тим, як відправити їх на сервер. Це економить трафік сервера, що означає менше навантаження на ваш сервер;
- Негайний відгук для відвідувачів – їм не потрібно чекати перезавантаження сторінки, щоб побачити, чи забули вони щось ввести;
- Цю мову легше вивчити, ніж інші мови програмування;
- Помилки простіше помітити і, отже, виправити;
- Він може бути призначений певним елементам веб-сторінки, або певним подіям, таким як кліки або миші;
- JS працює у багатьох браузерах, платформах, тощо;
- Використовувати JavaScript для перевірки введених даних та зменшення потреби в ручній перевірці даних;
- Можливість створювати інтерфейси, які реагують, коли користувач наводить на них мишу, або активує їх за допомогою клавіатури;
- Включення таких елементів, як компоненти перетягування та повзунки, щоб надати відвідувачам сайту більше зручності;
- Робить веб-сайти більш інтерактивними та утримує увагу відвідувачів;
- Більш швидкий і легший, ніж інші мови програмування.

Кожна мова програмування, яку можна придумати, має свої слабкі місця. Деякі недоліки включають:

- Вразливий до експлоїтів;
- Може використовуватися для виконання шкідливого коду на комп'ютері користувача;

- Не завжди підтримується різними браузерами та пристроями;
- Фрагменти коду JS можуть бути досить великими;
- Можуть відображатися по-різному на різних пристроях, що призводить до невідповідності в верстці;
- Клієнтський JavaScript не дозволяє читати чи записувати файли. Це обмеження існує з міркувань безпеки;
- JavaScript не можна використовувати для мережесх програм, оскільки такої підтримки немає;
- JavaScript не має функцій багатопотокової чи багатопроцесорної роботи.

Таким чином, хоча JavaScript був розроблен для того, щоб допомогти фронтенд-розробникам створювати інтерактивні елементи, можливості використання JavaScript швидко розширилися, включаючи такі речі, як розробка веб-сайтів, створення ігор та навіть розробка мобільних додатків.

2.4. Фреймворк Bootstrap

Bootstrap – це безкоштовний і відкритий CSS фреймворк, спрямований на інтерактивну мобільну розробку [20]. Він містить шаблони дизайну на основі CSS та (необов'язково) JavaScript для типографії, форм, кнопок, навігації та інших компонентів інтерфейсу.

Переваги фреймворку:

- Bootstrap – найпопулярніший фреймворк для створення макетів. Ось кілька додаткових причин використання Bootstrap:
 - CSS Bootstrap адаптується до телефонів, планшетів та настільних ПК;
 - Стили для мобільних пристроїв є головною метою фреймворку;
 - Bootstrap сумісний із усіма сучасними браузерами (Chrome, Firefox, Internet Explorer, Safari та Opera);
- Сітка Bootstrap – визначення спеціальних розмірів для кожного стовпця, використовуючи вбудовані невеликі, малі, середні, великі

					ІАЛЦ.045440.004 ПЗ	Арк.
						26
Зм	Лист	№ докум.	Підп.	Дата		

та надзвичайно великі розміри;

- Змінні розміри зображень – Bootstrap постачається з власним кодом для автоматичної зміни розміру зображень залежно від поточного розміру екрана. Просте додавання класу `.img-`, а заздалегідь визначені правила CSS подбають про інше. Також можна навіть змінювати форму зображень, додаючи такі класи, як `img-circle` та `img-rounded`;
- Вбудовані компоненти – Bootstrap постачається з цілою низкою компонентів, які можна легко застосувати до своєї веб-сторінки, включаючи:
 - Навігаційні панелі;
 - Випадаючі списки;
 - Стовпці прогресу;
 - Ескізи;
 - та інші;
- Bootstrap має велику спільноту та дружню підтримку;
- Bootstrap легко налаштувати і створити робочий макет менше ніж за годину;
- Вам не потрібно добре знати HTML і CSS, щоб використовувати Bootstrap, це плюс, якщо ви розробник бекенда і потрібно зробити деякі зміни в інтерфейсі користувача;
- Основна команда розробників фреймворку піклується про зворотну сумісність;
- Має чудову документацію;
- Компоненти Bootstrap добре прийняті до екосистеми популярних рамок JS MVC, таких як Angular;
- Bootstrap вирішує багато кросс-браузерних проблем, тому немає необхідності турбуватися про це більшу частину часу.

Як і більшість всього в житті, Bootstrap має свої мінуси:

- Файли Bootstrap можуть бути занадто великими, враховуючи функціональність, яку пропонує цей фреймворк. Це може призвести до збільшення часу завантаження веб-сайтів, особливо у повільних мережах. Початківцям може бути важко визначити та виправити цю проблему; однак, існують інструменти налаштування, які можуть допомогти усунути зайвий код для функцій, які ніколи не використовуються.
- Bootstrap може перешкоджати фактичному вивченню програмування – завжди є ризик, що, використовуючи Bootstrap, можна потрапити в цикл простого копіювання існуючого коду, не розуміючи його. Витративши час, щоб дійсно дізнатися, що ви робите, можна використовувати Bootstrap як спосіб прискорити навчання, а не перешкоджати йому.

Висновки до розділу

В даному розділі було проведено короткий огляд технологій, які будуть використовуватись при розробці редактору. Взагалі, при веб-розробці майже завжди використовуються три технології – HTML, CSS та JavaScript. Також існує безліч сторонніх фреймворків, які полегшують працю при розробці різних компонентів та при створенні адаптивності сайтів. Наприклад, фреймворк Bootstrap дозволяє усього цього досягнути, тому його і було обрано для використання, разом з вище зазначеними технологіями, при розробці редактору для створення симетричних зображень.

3. РОЗРОБКА РЕДАКТОРУ ДЛЯ СТВОРЕННЯ СИМЕТРИЧНИХ ЗОБРАЖЕНЬ

3.1. Огляд структурних елементів редактору

Розглянемо головне вікно редактора (Рис 3.1):

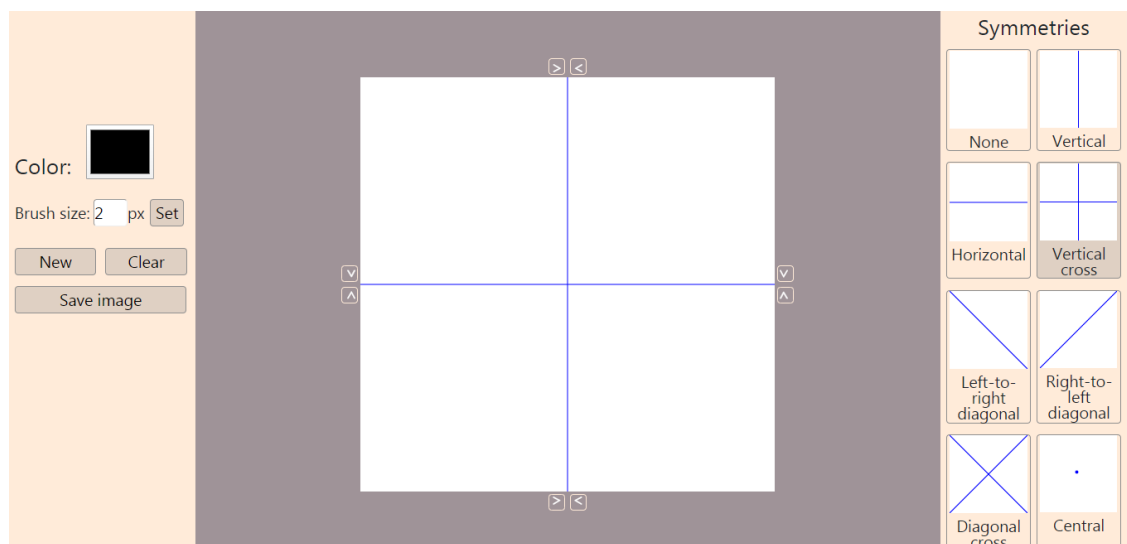


Рис 3.1 – Головне вікно редактора

Редактор складається з трьох основних елементів: бокових панелей та центральної панелі. Кожна з цих панелей містить у собі елементи, які відповідають за ті чи інші функції програми.

Спочатку розглянемо ліву панель (Рис 3.2):

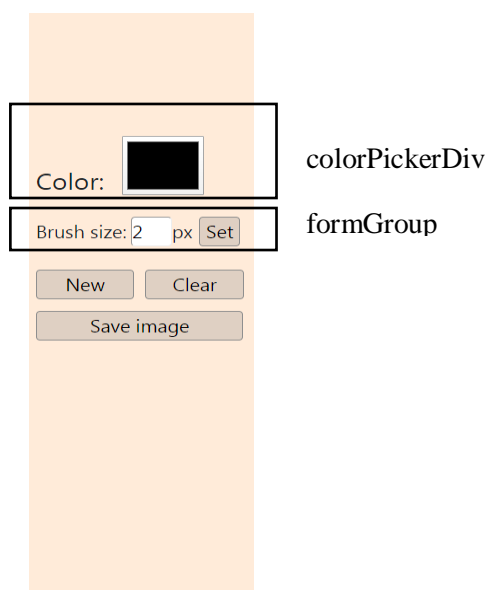


Рис 3.2 – Ліва панель редактора

Ширина цього елемента складає дві колонки, якщо брати до уваги розмітку фреймворку Bootstrap, у якого загальна ширина усього екрану складає дванадцять колонок. Ліва панель відповідає за наступні функції редактора:

- Вибір кольору пензля;
- Зміна розміру пензля;
- Створення нового зображення;
- Видалення холсту;
- Зберегти зображення.

Великий прямокутник елемента `colorPickerDiv` показує поточне значення кольору пензля. Натискання на цей прямокутник призведе до виклику інструменту вибору кольору, у якому користувач може вибрати якийсь інший колір, таким чином замінивши поточний.

У полі встановлення розміру пензля користувач може ввести бажане значення розміру, а натиснувши на кнопку «Set» встановити його. Значення розміру, яке вводить користувач, контролюється регулярним виразом: введені значення повинні бути тільки додатнім числом, не менше 1 та не більше 300 пікселів. Обмеження на максимальний розмір встановлене для того, щоб виключити пензлі занадто великих розмірів, які можуть бути більше ніж поточний розмір холсту. У разі неправильно введених даних, користувач отримає вікно з попередженням, а поточний розмір пензля ніяким чином не зміниться (Рис 3.3).

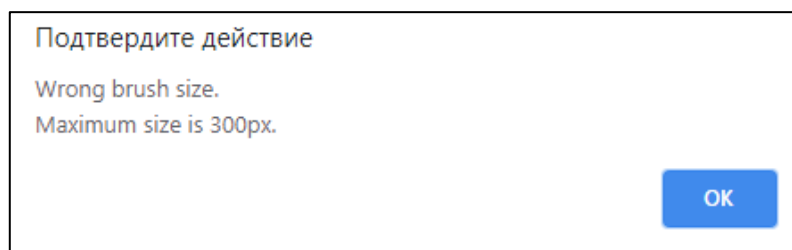


Рис 3.3 – Помилка при неправильно введеному розмірі пензля

Натискання на кнопку «New» створює новий холст. Перед створенням, з'явиться вікно підтвердження дії (Рис 3.4), у якому буде сказано, що

незбережене зображення буде видалене та замінено новим пустим:

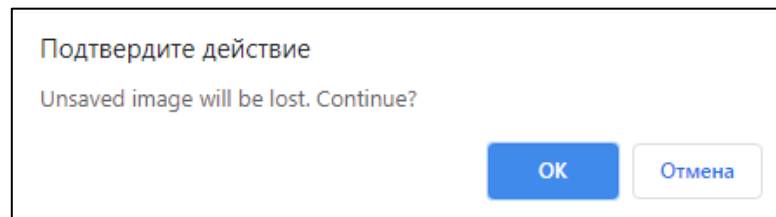


Рис 3.4 – Підтвердження створення нового зображення

Якщо користувач підтверджує свою дію, то старий холст видаляється, на його місце створюється новий, зберігаючи вибрану симетрію малювання.

Кнопка «Clear» також чекає підтвердження користувача на видалення поточного зображення і, у разі згоди, також створює новий холст із збереженням вибраної симетрії.

Ну і наостанок, кнопка «Save image» зберігає поточне зображення на холсті у вигляді файлу формату .png.

Перш ніж перейти до центральної панелі, розглянемо праву бокову (Рис 3.5):

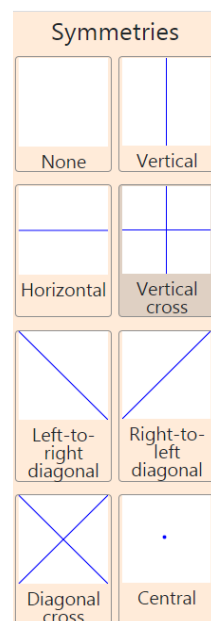


Рис 3.5 – Права панель редактора

Ширина цієї панелі також складає дві колонки, беручи до уваги розмітку фреймворку Bootstrap. Права панель відповідає за вибір та зміну поточного типу симетрії, який буде встановлено при малюванні. Редактор дає змогу обрати будь-

яку симетрію з восьми типів, а саме:

- Пуста – користувач може малювати як завгодно без осей симетрії;
- Вертикальна – лінії відображаються симетрично від осі Y;
- Горизонтальна – лінії відображаються симетрично від осі X;
- Вертикальна хрестова – лінії відображаються симетрично як від осі Y, так і від осі X;
- Діагональна зліва-направо – лінії відображаються симетрично відносно діагоналі холсту, яка починається в верхньому лівому куті та закінчується в правому нижньому;
- Діагональна зправа-наліво – лінії відображаються симетрично відносно діагоналі холсту, яка починається в верхньому правому куті та закінчується в лівому нижньому;
- Діагональна хрестова – лінії відображаються симетрично відносно діагоналей холсту;
- Центральна – лінії відображаються симетрично відносно центра холста.

Тепер перейдемо до розгляду центральної панелі (Рис. 3.6):

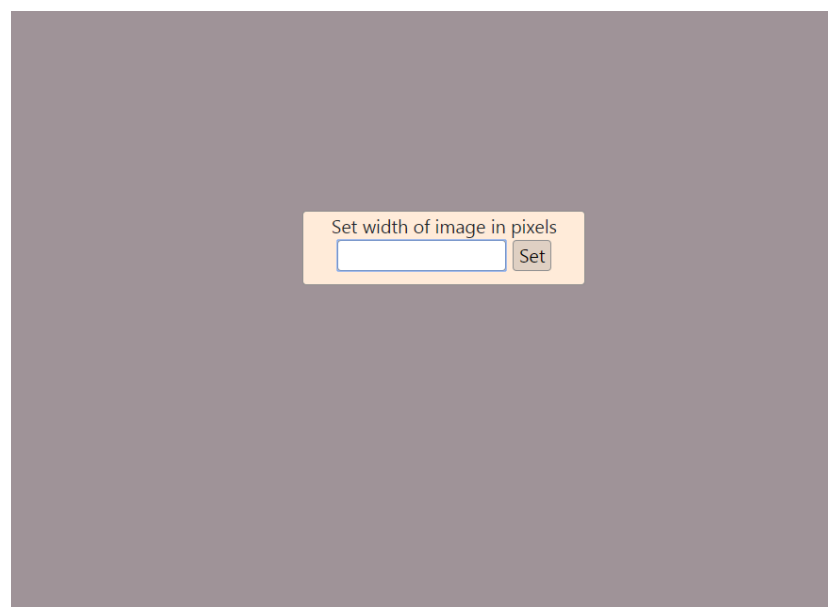


Рис. 3.6 – Центральна панель при виборі розміру холсту

Ширина цієї панелі складає вісім колонок, беручи до уваги розмітку фреймворку Bootstrap. У сумі всі три панелі як раз займають дванадцять колонок,

а саме дві бокові панелі по дві колонки кожна, та центральна шириною у вісім.

При початку роботи з редактором, центральна панель має лише поле для введення бажаного розміру холсту. Значення, яке вводиться в це поле, також, як і у разі з полем встановлення розміру пензля, контролюється регулярним виразом, який перевіряє коректність введеної інформації. Введене значення повинно бути тільки додатнім числом, не менше 100 пікселів та не більше висоти вікна. Це обмеження встановлене для того, щоб виключити створення занадто маленьких розмірів холстів, або занадто великих, які не поміщаються на екрані. Як і у разі з полем встановлення розміру пензля, якщо користувач введе неправильні дані, то з'явиться вікно з попередженням, при чому ніякого холста при цьому створено не буде (Рис. 3.7):

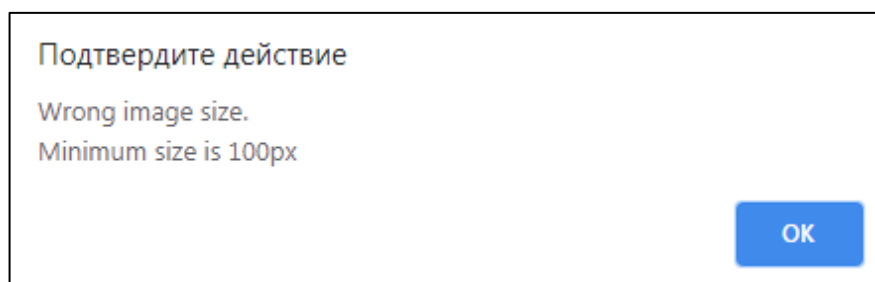


Рис 3.7 – Попередження при неправильному розмірі створюваного холсту

Якщо ж користувач надав коректні дані, то на центральній панелі буде створено основний холст (Рис 3.8):

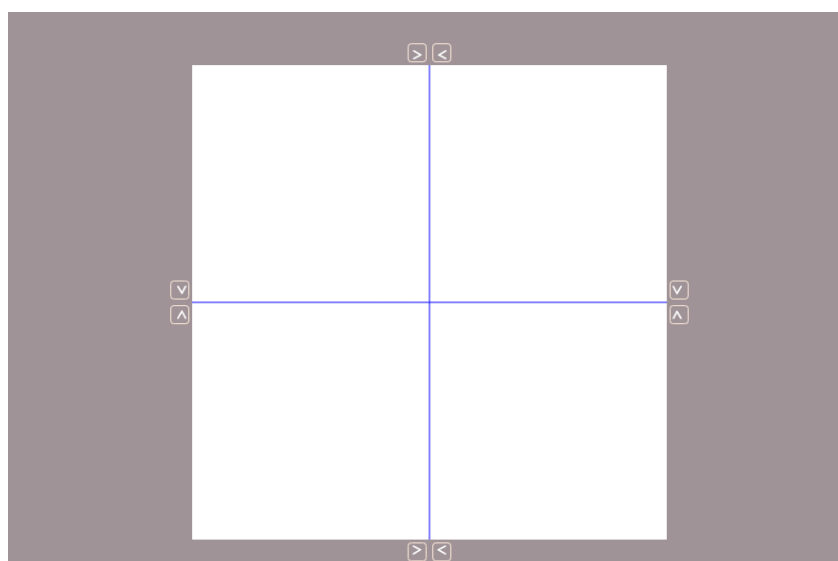


Рис. 3.8 – Центральна панель зі створеним холстом

Насправді, при цьому використовується два холсти, один поверх іншого: нижній є основним холстом, на якому користувач і малює зображення, а на верхньому зображено поточну симетрію для наочності. Через цю накладенність немає змоги напряму дістатися до нижнього основного холсту, тому всі координати переміщення курсору реєструються обробником рухів мишки верхнього холсту, а потім вже ці координати використовуються при малюванні зображення на нижньому основному холсті.

3.2. Малювання ліній

Ключовою необхідністю майже будь-якого графічного редактору є малювання (прямої) лінії на растровому пристрої. Звісно, ми хочемо, щоб вибраний нами алгоритм був настільки швидким, наскільки можливо, тому що на практиці цей алгоритм буде використаний дуже велику кількість разів.

Існує декілька алгоритмів, які використовуються як раз для цього, тому розглянемо основні з них.

Побудова прямих за алгоритмом Брезенхама

Цей алгоритм був розроблений Брезенхамом [21]. Це ефективний метод малювання ліній, оскільки включає лише цілі операції додавання, віднімання та множення. Ці операції виконуються дуже швидко, тому лінії також можна генерувати з великою швидкістю. У цьому методі наступним пікселем для генерації, вибирається той, хто має найменшу відстань від істинної лінії.

Переваги:

- Передбачає лише цілу арифметику, тому це швидко;
- Уникає утворення повторюваних точок;
- Може бути реалізований за допомогою апаратних засобів, оскільки не використовує множення та ділення;
- Швидший порівняно з ЦДА (цифровим диференціальним аналізатором), оскільки не передбачає обчислень з плаваючою комою, як алгоритм ЦДА, який бути розглянуто пізніше.

Недоліки:

- Цей алгоритм призначений лише для малювання базових ліній. Ініціалізація не є частиною лінійного алгоритму Брезенхама. Отже, щоб намалювати плавні лінії, треба розглянути інший алгоритм, наприклад, алгоритм Сяоліна Ву, який також бути розглянуто пізніше.

Алгоритм Брезенхама [22]:

- Крок 1: Запустіть алгоритм;
- Крок 2: оголосити змінні $x_1, x_2, y_1, y_2, d, i_1, i_2, dx, dy$;
- Крок 3: Введіть значення x_1, y_1, x_2, y_2
Де x_1, y_1 – координати початкової точки,
 x_2, y_2 – координати точки закінчення;
- Крок 4: Обчисліть $dx = x_2 - x_1$
Обчисліть $dy = y_2 - y_1$
Обчисліть $i_1 = 2 * dy$
Обчисліть $i_2 = 2 * (dy - dx)$
Обчисліть $d = i_1 - dx$;
- Крок 5: Розгляньте (x, y) як вихідну точку і $x_{\text{кінцеве}}$ як максимально можливе значення x .
Якщо $dx < 0$
Тоді $x = x_2$
 $y = y_2$
 $x_{\text{кінцеве}} = x_1$
Якщо $dx > 0$
Тоді $x = x_1$
 $y = y_1$
 $x_{\text{кінцеве}} = x_2$
- Крок 6: Створіть точку за координатами (x, y) ;

- Крок 7: Перевірте, чи генеровано цілий рядок.

Якщо $x \geq x_{\text{кінцеве}}$

Стоп.

- Крок 8: Обчисліть координати наступного пікселя

Якщо $d < 0$

Тоді $d = d + i_1$

Якщо $d \geq 0$

Тоді $d = d + i_2$

Інкремент $y = y + 1$

- Крок 9: Інкремент $x = x + 1$;
- Крок10: Накресліть точку останніх (x, y) координат;
- Крок11: Перейдіть до кроку 7;
- Крок12: Кінець алгоритму.

Нехай нам необхідно побудувати лінію з точки з координатами $(0;0)$ в точку з координатами $(6;2)$ використовуючи алгоритм Брезенхама (Рис. 3.9):

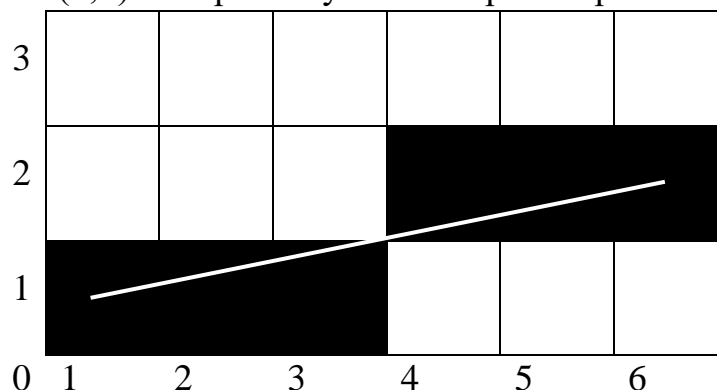


Рис. 3.9 – Пряма, створена за алгоритмом Брезенхама

Побудова прямих за алгоритмом цифрового диференціального аналізатора

Це поступовий метод перетворення лінії [23]. У цьому методі розрахунок проводиться на кожному кроці, але за допомогою результатів попередніх етапів.

Переваги:

- Це швидший метод, ніж метод використання прямого використання рівняння прямої;

- Цей метод не використовує множення;
- Це дозволяє нам виявити зміну значень x і y , тому побудувати одну і ту ж точку двічі неможливо;
- Цей метод дає індикацію переповнення, коли точка переставляється;
- Це простий метод, оскільки кожен крок включає лише два додавання.

Недоліки:

- Передбачає завершення додавання з плаваючою комою. Накопичення помилки округлення викликає накопичення помилки;
- Округлення та операції з плаваючою точкою вимагають багато часу;
- Більше підходить для генерації ліній за допомогою програмного забезпечення. Але він менш підходить для апаратної реалізації.

Алгоритм DDA [24]:

- Крок 1: Запустіть алгоритм;
- Крок 2: Оголосити $x_1, y_1, x_2, y_2, dx, dy, x, y$ як цілі змінні;
- Крок 3: Введіть значення x_1, y_1, x_2, y_2

Де x_1, y_1 – координати початкової точки,
 x_2, y_2 – координати точки закінчення;

- Крок 4: Обчисліть $dx = x_2 - x_1$;
- Крок 5: Обчисліть $dy = y_2 - y_1$;
- Крок 6. Якщо $ABS(dx) > ABS(dy)$

Тоді $step = abs(dx)$

Інакше

- Крок 7: $x_{інк} = dx / step$

$y_{інк} = dy / step$

призначити $x = x_1$

призначити $y = y_1$;

- Крок 8: Встановлення пікселя (x, y) ;

- Крок 9: $x = x + x_{\text{інк}}$

$$y = y + y_{\text{інк}}$$

Встановити пікселі (округлити (x), округлити (y));

- Крок10: Повторіть крок 9, поки $x \neq x_2$;
- Крок11: Закінчення алгоритму.

Побудова прямих за алгоритмом Ву

Нерівні лінії є головними перешкодами для досягнення професійних проявів растрової графіки. Антиаліасінг(англ. «antialiasing», або можна сказати згладжування) може створювати дуже плавні лінії та надавати їм стильний зовнішній вигляд.

Растрезація – це завдання взяти зображення, описане у форматі векторної графіки (фігури), і перетворити його в растрове зображення (серію пікселів, крапок, або ліній, які при відображенні разом створюють зображення, яке було представлене через формив векторній графіці). Потім растроване зображення може відображатися на дисплеї комп'ютера, відеодисплея, принтера, або зберігатися у форматі растрових файлів.

Якість растрезації можна покращити за допомогою антиаліасінгу, який створює "гладкі" краї. Ми також іноді говоримо про точність пікселів, яка враховує позиції в більш меншому масштабі, ніж піксельна сітка, і може давати різні результати, навіть якщо кінцеві точки примітиву потрапляють у однакові піксельні координати, створюючи плавніші анімації руху.

У комп'ютерній графіці антиаліасінг покращує зовнішній вигляд "зазуброваних" країв багатокутника, або "зубців", тому їх згладжують на екрані. Однак ця технологія вимагає плати за продуктивність для відеокарти та використовує більше відеопам'яті. Рівень згладжування визначає, наскільки гладкими є краї полігону (і скільки відеопам'яті антиаліасінг споживає).

Для швидкого антиаліасінгу Сяолін Ву винайшов алгоритм, названий його іменем: Ву Антиаліасінг [25]. Для прямої з координатами кінцевих точок (0;0) та (6;2) алгоритм Ву дасть наступний результат (Рис. 3.10):

					ІАЛЦ.045440.004 ПЗ	Арк.
						38
Зм	Лист	№ докум.	Підп.	Дата		

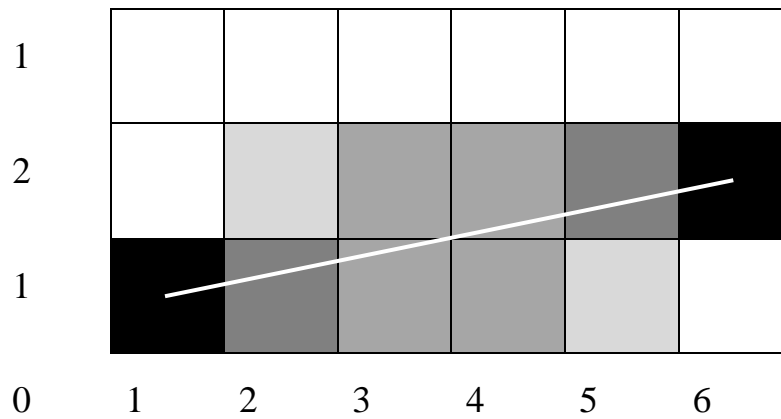


Рис. 3.10 – Пряма, створена за алгоритмом Ву

Лінійний алгоритм Сяоліня Ву характеризується тим, що на кожному етапі обчислення проводяться для двох найближчих до лінії пікселів, і вони забарвлюються з різною інтенсивністю, залежно від відстані до шуканої «ідеальної» прямої. Інтенсивність поточного перетину середнього пікселя дає 100% забарвлення, якщо піксель знаходиться в межах 0,9 пікселя, інтенсивність буде 10%. Іншими словами, сто відсотків інтенсивності ділиться між пікселями, які обмежують векторну лінію з обох сторін.

У розроблюваному редакторі також використовується стандартний метод елементу «canvas», а саме `canvas.lineTo()`. У якості початкових та кінцевих точок прямої використовуються координати руху миші користувача на холсті. Результат малювання лінії пензлем товщиною 2px (Рис. 3.11):

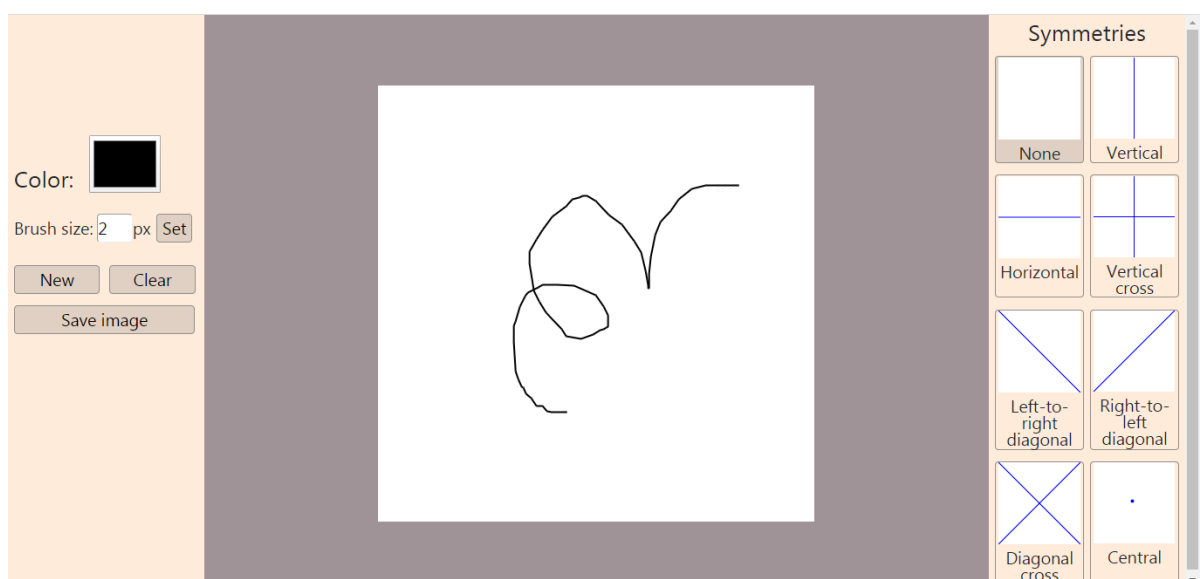


Рис. 3.11 – Малювання ліній пензлем товщиною 2px

Але при використанні розмірів пензля більше ніж 5px, починають проявлятися недоліки цього методу (Рис. 3.12), наприклад при малюванні пензлем розміром 30px:

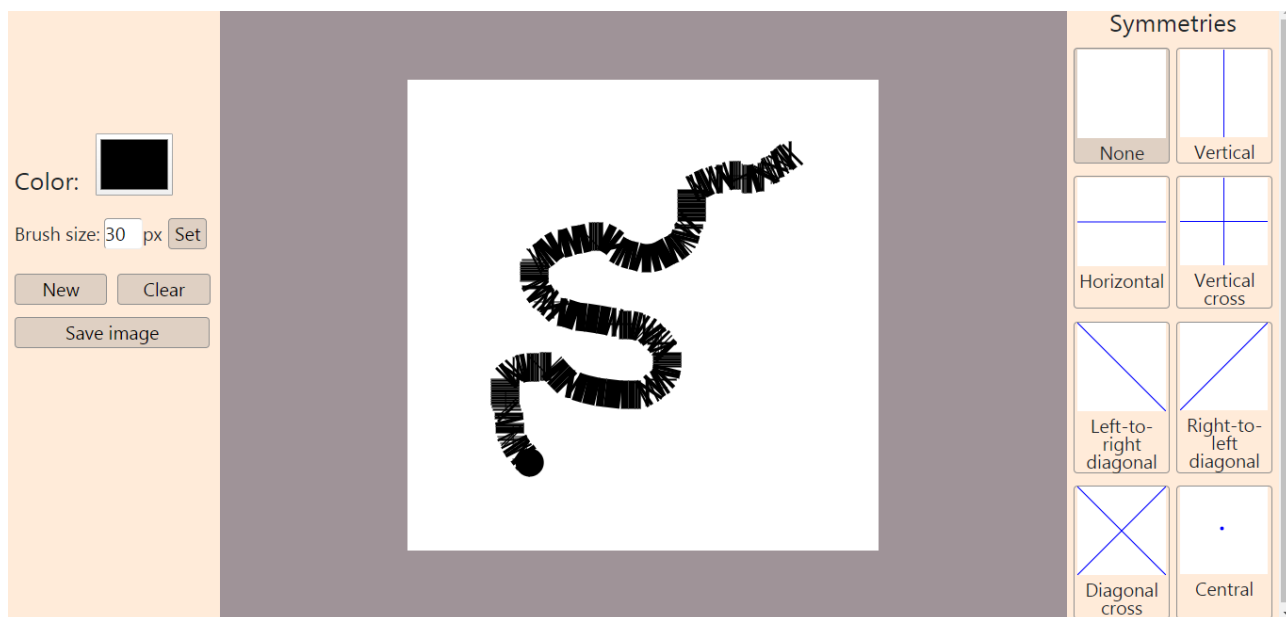


Рис. 3.12 – Цікаві ефекти при малюванні пензлем, розмір якого більше 5px

Звичайно, можна сказати, що «це не баг, а фіча», та зробити це додатковим ефектом пензля в наступних версіях програми. Але на даний момент це руйнує використання пензлів розміром більше 5px. Тому при малюванні пензлями більше 5px, в розробленому редакторі використовується алгоритм малювання прямої з використанням кіл, тому розглянемо його детальніше на прикладі коду з програми. Функція, яка відповідає за це, має назву `draw` та приймає 7 параметрів:

- `id` холсту, на якому треба малювати;
- початкові координати лінії;
- кінцеві координати лінії;
- розмір пензля в пікселях;
- поточний колір пензля.

Усі ці параметри будуть використовуватись при малюванні ліній, розмір яких перевищує 5px.

Код самої функції:

```
function draw(canv_id, startX, startY, endX, endY, lineWidth, color) {
    var ctx = document.getElementById(canv_id).getContext("2d");
    var distance = Math.sqrt((endX - startX) * (endX - startX) +
                             (endY - startY) * (endY - startY));
    var radius = lineWidth / 2;
    var steps = distance / (radius / 2);
    var deltaX = (endX - startX) / steps;
    var deltaY = (endY - startY) / steps;
    for(i = 0, x = startX, y = startY; i < steps; i++, x += deltaX,
                                             y += deltaY) {

        ctx.beginPath();
        ctx.arc(x, y, radius, 0, Math.PI+(Math.PI*2)/2, true);
        ctx.fillStyle = color;
        ctx.fill();
        ctx.closePath();
    }
}
```

Спочатку в змінну ctx заноситься посилання на поточний холст, на якому власне і буде створена пряма. Потім вираховується відстань від точки з початковими координатами до кінцевої точки. Для цього застосовується теорема Піфагора: якщо розглядати дві кінцеві точки як кінці гіпотенузи прямокутного трикутника, то різниця між їх x-координатами буде довжиною першого катета, а різниця між їх y-координатами буде довжиною другого катета. Після цього довжина відстані дорівнює квадратному кореню з суми квадратів цих двох катетів.

Радіус кола, яке буде побудовано, дорівнює половині розміру пензля. Змінна steps містить у собі кількість кроків, за які буде побудовано шукану пряму. У даному випадку її значення дорівнює відстані між кінцевими точками прямої поділеної на розмір радіус кола та знов поділеної на 2. Ділити можна на будь-яке число більше 2, це просто означає, що кола будуть розташовані більш

					ІАЛЦ.045440.004 ПЗ	Арк.
						41
Зм	Лист	№ докум.	Підп.	Дата		

близько одне до одного. Змінні ΔX та ΔY містять значення різниці в координатах, на які буде зміщено центр поточного кола на кожному кроці. Ці значення можуть бути як додатніми, так і від'ємними, це просто означає в яку сторону буде пересуватися центр – в разі з x -координатами додатні значення це зсув вправо, а від'ємні – вліво. Для y -координат додатні значення це зсув вниз, а від'ємні – зсув вверх.

Загальний цикл малювання використовує значення змінної $steps$, як необхідну кількість кроків. Таким чином від'ємні значення змінних ΔX та ΔY ніяким чином не впливають на кількість кроків виконання циклу. Створення першого кола починається в точці з початковими координатами. Тут тільки малюється коло діаметром з розмір пензля та заливається поточним кольором пензля. Після цього змінюються поточні координати центра кола, і цей процес продовжується, доки не буде намальована пряма, що вимагається. Результат роботи цього алгоритму представлений на (Рис. 3.13) з лінією близької до той, що намальована на (Рис. 3.12):

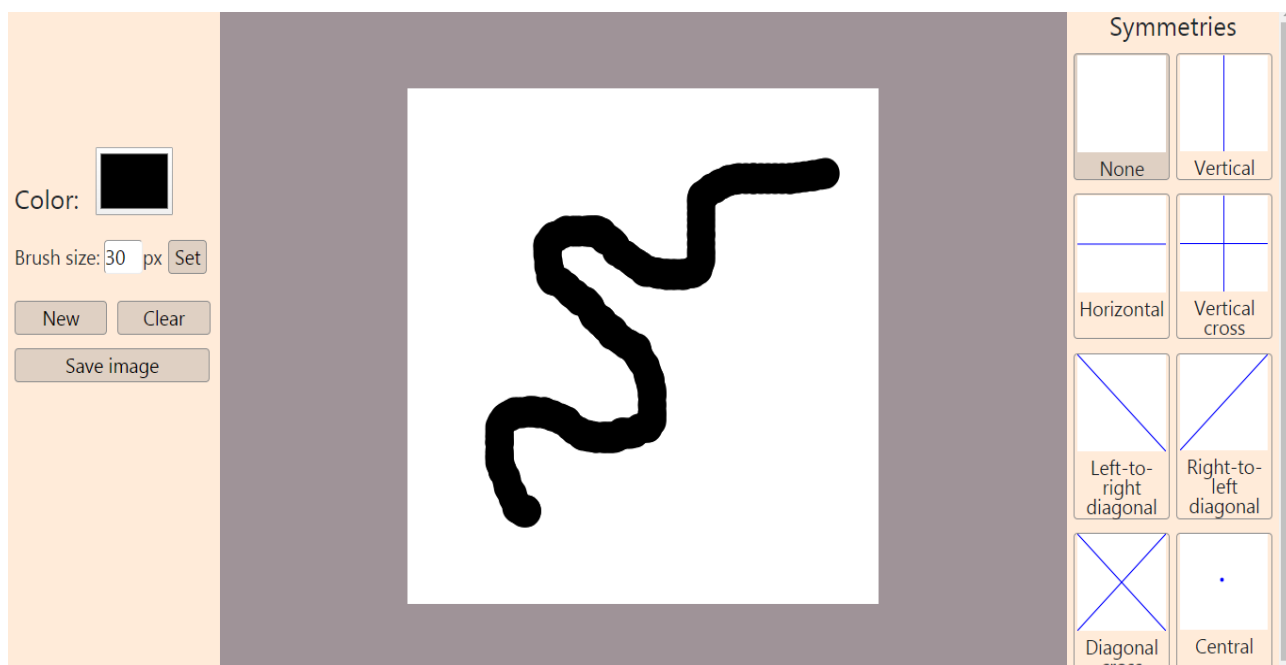


Рис. 3.13 – Використання алгоритму з малюванням кіл призводить до більш гарних результатів при великих розмірах пензля

3.3. Перетворення координат та малювання при кожному типі симетрії

Розглянемо на наступному прикладі перетворення координат при кожному типі симетрії (Рис 3.14) (параметр $size$ – це розмір/довжина/ширина холсту):

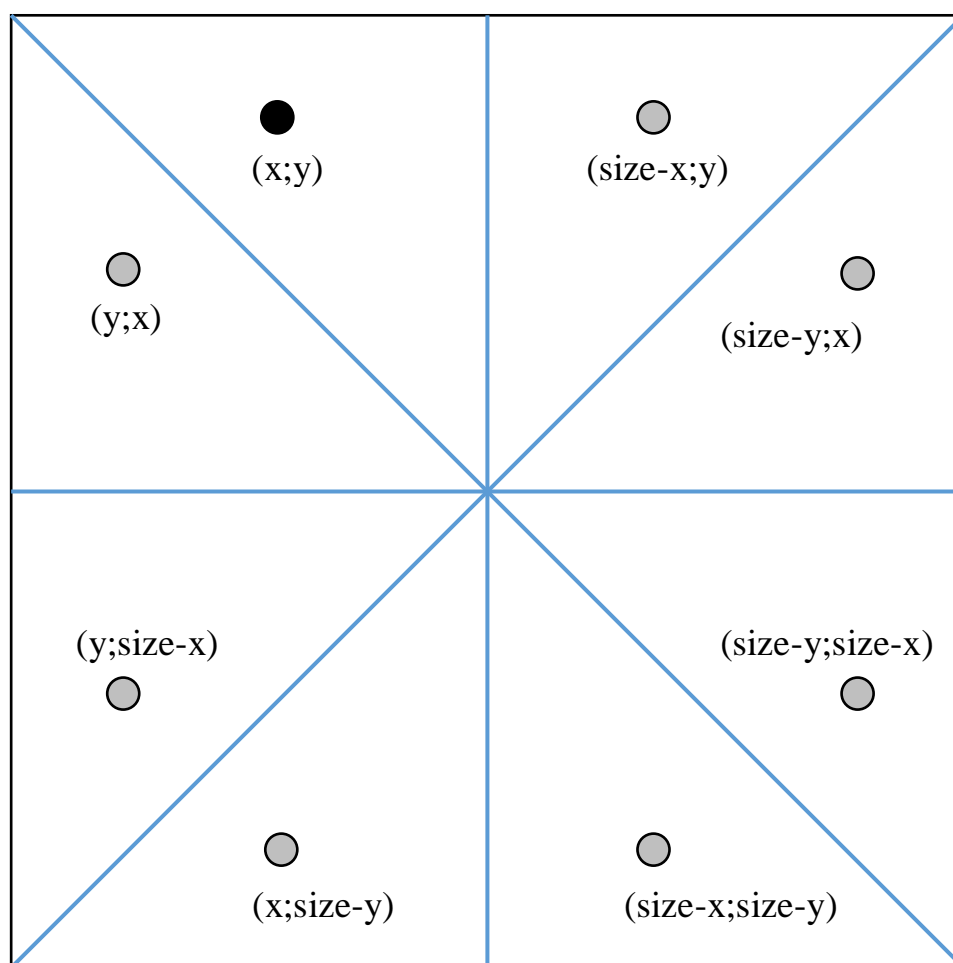


Рис 3.14 – Перетворення координат та малювання при кожному типі симетрії

Нехай початкова точка має координати $(x; y)$. При обговоренні кожного типу симетрії, будемо вважати, що точка з початковими координатами завжди малюється за замовчуванням, використовуючи її поточні координати. Це має сенс, оскільки не зважаючи на вибраний тип симетрії, користувач повинен мати змогу завжди малювати лінії там, де в даний момент часу знаходиться курсор миші. Тоді для наступних типів симетрій:

- Пуста – перетворення не потрібні, лінії малюються лише для поточних координат пензля;
- Вертикальна – лінія, що малюється, відображається симетрично від осі Y ,

додаткова точка, при заданій симетрії має координати (size - x; y);

- Горизонтальна – лінія, що малюється, відображається симетрично від осі X додаткова точка, при заданій симетрії має координати (x; size - y);
- Вертикальна хрестова – це випадок з одночасним використанням вертикальної, горизонтальної та центральної симетрій. Лінії відображаються симетрично як від осі Y, від осі X, так і відносно центру холста. Тому маємо додаткові три точки. Дві з них це випадки вертикальної та горизонтальної симетрій – (size - x; y) та (x; size - y), а третя має центральну симетрію – (size - x; size - y);
- Діагональна зліва-направо – лінія, що малюється, відображається симетрично від діагоналі холста, яка починається в верхньому лівому куті та закінчується в правому нижньому. Координати додаткової точки просто інверсуються: (y; x);
- Діагональна зправа-наліво – лінія, що малюється, відображається симетрично від діагоналі холста, яка починається в верхньому правому куті та закінчується в лівому нижньому. Координати додаткової точки інверсуються та відраховуються відносно розміру холста: (size - y; size - x);
- Діагональна хрестова – це випадок з одночасним використанням симетрій відносно двох діагоналей холсту та центральної симетрії. Лінії відображаються симетрично як від обох діагоналей, так і відносно центру холста. Тому маємо додаткові три точки. Дві додаткові точки відносно двох діагоналей: (y; x) та (size - y; size - x), а також третю з центральною симетрією – (size - x; size - y);
- Центральна – лінія, що малюється, відображається симетрично відносно центру зображення, ніби ці дві точки утворюють діаметр кола з центром в центрі холста. Додаткова точка має координати відносно розмірів холста, а саме (size - x; size - y).

3.4. Модифікатори симетрії

Ключовою особливістю розроблюваного редактору є можливість малювання не зовсім симетричних зображень використовуючи модифікатори симетрії та не змінюючи при цьому поточної симетрії.

Розглянемо це на прикладі з використанням однієї та двох осей симетрії. Спочатку розглянемо випадок з однією (Рис. 3.15):

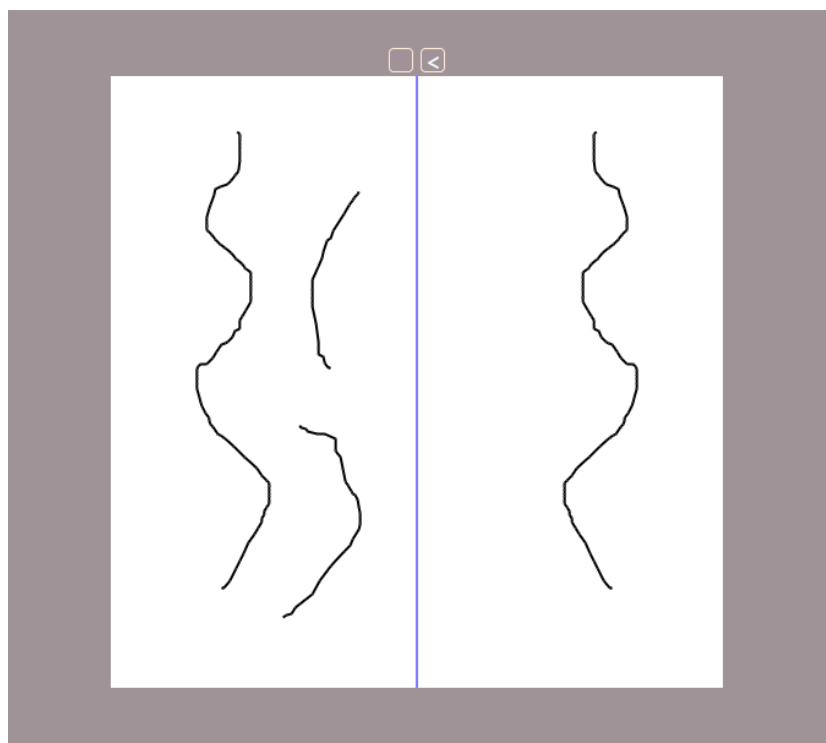


Рис 3.15 – Використання модифікаторів симетрії з однією віссю симетрії

Ось ці кнопки зі стрілочками і є власне модифікаторами симетрії. Якщо на них зображена стрілка, то це означає, що при малюванні в сегменті із якого ця стрілка вказує, зображення буде перенесене в сегмент на який вона вказує. Як можна побачити на прикладі, при малюванні в правій частині холсту, зображення було перенесене на ліву частину. У той час, модифікатор лівої частини був вимкнений, тому малювання в лівій частині не дало ніякого впливу на праву частину.

До симетрій, які також використовують лише одну вісь симетрії також відносяться горизонтальна та дві діагональних. Принцип їх дії ідентичний до щойно розглянутого.

Більше уваги заслуговують два види симетрії з використанням двох осей симетрії, а саме вертикальна хрестова та діагональна хрестова. Складність роботи з ними полягає в тому, що дві осі симетрії розбивають холст на чотири сегменти, а не на два. Розглянемо це на прикладі вертикальної хрестової симетрії (Рис. 3.16):

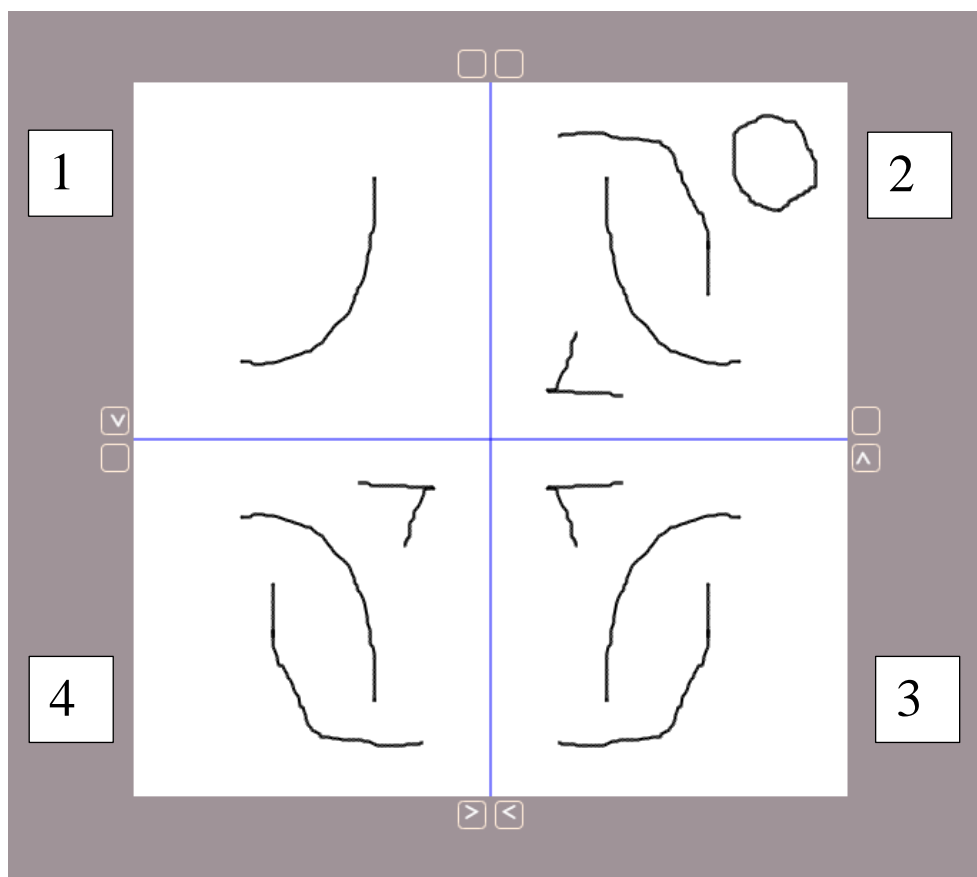


Рис. 3.16 – Використання модифікаторів симетрії з двома осями симетрії

Нумерація сегментів починається з верхнього лівого кута та йде за годинниковою стрілкою. Як можна побачити з рисунку, малювання прямої в першому сегменті призводить до його віддзеркалення в усіх чотирьох сегментах, хоча перенос симетрії з сегменту №1 в сегмент №2 вимкнено. Це відбувається через те, що віддзеркалення спочатку переходить в четвертий сегмент, з нього в третій, і, нарешті, у другий; і таким чином усі чотири сегменти мають спільну намальовану лінію.

Лінія в сегменті №4 відображається в сегмент №3, а з нього в сегмент №2. Під час цього також відбувається послідовне відображення. Аналогічно для

сегменту №3, зображення якого переходить у другий та четвертий сегменти.

Малювання в сегменті №2 не призводить до жодних змін в усіх сегментах, оскільки обидва модифікатори симетрії цього сегменту вимкнені.

Також треба зауважити, що малювання в другому, третьому та четвертому сегментах не призводить до змін в першому сегменті, тому що переноси з суміжних з першим сегментів (модифікатори другого та четвертого сегментів на перенос в перший) вимкнені.

Висновки до розділу

В даному розділі було розглянуто елементи редактора для створення симетричних зображень та принцип їх роботи. Редактор складається з трьох основних елементів: бокових та центральної панелі. На центральній панелі розміщується основний холст та холст симетрії. Особливістю редактора є зміна алгоритму малювання, якщо поточний розмір пензля перевищує п'ять пікселів, що дозволяє уникнути «обтріскування» лінії, що малює пензель. Ключовою відмінністю створеної програми від існуючих аналогів є використання так званих модифікаторів симетрії, які дозволяють створювати будь-які типи зображень не змінюючи при цьому вибраний тип симетрії.

ВИСНОВКИ

Метою даного дипломного проєкту було розроблення редактору для створення симетричних зображень. Цей редактор також має змогу використовувати модифікатори симетрії, які дозволяють користувачам створювати симетричні зображення з вимкненням відображення на будь-яку частину, не змінюючи при цьому вибрану симетрію.

В результаті аналізу існуючих рішень було визначено їх невеликий недолік: неможливість вимкнення відображення на будь-яку частину від осі симетрії. Через це користувачам доводиться взагалі вимикати дзеркальний режим, щоб домалювати якусь несиметричну деталь. Проте наявність цієї опції дозволила б уникнути цих незручностей. Актуальність створення симетричних зображень є дуже великою, наприклад, при малюванні обличчь, тварин, інструментів, проектуванні схем та креслень, персонажів, предметів, будівель, будівництві кораблів, або літальних апаратів.

Розроблений редактор дозволяє:

- встановлювати бажаний розмір зображення;
- змінювати колір та розмір пензля;
- малювати зображення;
- вибирати бажаний тип симетрії;
- створювати та зберігати зображення.

Редактор був реалізований у вигляді веб-сторінки, для полегшення доступу для користувачів. Так як майже у всіх користувачів є доступ до мережі Інтернет, і не всі користувачі можуть мати змогу чи бажання для встановлення якихось додаткових програм на їх комп'ютери.

При розробці редактору були використані HTML/CSS, мова програмування JavaScript та фреймворк Bootstrap 4.

В подальшому розроблений редактор може бути доповнений новими функціями, наприклад, відміною дії, зчитуванням картинки з файлу формату

					ІАЛЦ.045440.004 ПЗ	Арк.
						48
Зм	Лист	№ докум.	Підп.	Дата		

.png, або .jpg та встановленням її у якості зображення для редагування на холсті, можливістю створювати прямі лінії, а також «гарячими» клавішами для пришвидшення роботи та покращення взаємодії користувача з програмою. Або цей редактор може стати плагіном для якогось існуючого графічного редактору, чи взагалі стати новим графічним редактором.

					ІАЛЦ.045440.004 ПЗ	Арк.
						49
Зм	Лист	№ докум.	Підп.	Дата		

СПИСОК ВИКОРИСТАНИХ ЛІТЕРАТУРНИХ ДЖЕРЕЛ

1. Огляд і класифікація сучасних графічних редакторів. Формати графічних файлів. – Електрон. дані (1 файл) – Режим доступу: http://wiki.fizmat.tnpu.edu.ua/index.php/Огляд_і_класифікація_сучасних_графічних_редакторів_Формати_графічних_файлів. (дата звернення 11.05.2020) – Назва з екрану.
2. Графічні редактори. – Електрон. дані (1 файл) – Режим доступу: https://stud.com.ua/54389/informatika/grafichni_redaktori (дата звернення 11.05.2020) – Назва з екрану.
3. История создания Adobe Photoshop. – Електрон. дані (1 файл) – Режим доступу: <https://akvis.com/ru/articles/photoshop-history/index.php> (дата звернення 11.05.2020) – Назва з екрану.
4. Photoshop. – Електрон. дані (1 файл) – Режим доступу: <https://whatis.techtarget.com/definition/Photoshop> (дата звернення 11.05.2020) – Назва з екрану.
5. Вибір параметрів симетрії під час малювання. – Електрон. дані (1 файл) – Режим доступу: <https://helpx.adobe.com/ua/photoshop/using/paint-symmetry.html> (дата звернення 11.05.2020) – Назва з екрану.
6. Обзор 3ds max: кратко о главном. – Електрон. дані (1 файл) – Режим доступу: <https://3ddevice.com.ua/blog/3d-printer-obzor/obzor-3ds-max/> (дата звернення 11.05.2020) – Назва з екрану.
7. Symmetry Modifier. – Електрон. дані (1 файл) – Режим доступу: <https://knowledge.autodesk.com/support/3ds-max/learn-explore/caas/CloudHelp/cloudhelp/2019/ENU/3DSMax-Modifiers/files/GUID-EB0B7B9B-117D-4CC3-A966-A3E007E0C68A-htm.html> (дата звернення 11.05.2020) – Назва з екрану.
8. CorelDraw. – Електрон. дані (1 файл) – Режим доступу: <https://uk.wikipedia.org/wiki/CorelDraw> (дата звернення 12.05.2020) – Назва з екрану.

					ІАЛЦ.045440.004 ПЗ	Арк.
Зм	Лист	№ докум.	Підп.	Дата		50

з екрану.

9. Creating symmetry. – Електрон. дані (1 файл) – Режим доступу: <https://learn.corel.com/tips/graphics/creating-symmetry/> (дата звернення 12.05.2020) – Назва з екрану.
- 10.Що таке веб сайт?. – Електрон. дані (1 файл) – Режим доступу: <https://futurenow.com.ua/shho-take-veb-sajt-yaka-istoriya-yih-vynyknennya-ta-vydy-veb-sajtiv-shho-take-veb-storinka/> (дата звернення 12.05.2020) – Назва з екрану.
- 11.История развития HTML. – Електрон. дані (1 файл) – Режим доступу: https://vertex-academy.com/tutorials/ru/html_history/ (дата звернення 12.05.2020) – Назва з екрану.
- 12.World Wide Web Consortium (W3C). – Електрон. дані (1 файл) – Режим доступу: <https://www.w3.org/> (дата звернення 12.05.2020) – Назва з екрану.
- 13.Що таке CSS. – Електрон. дані (1 файл) – Режим доступу: https://css.in.ua/article/shcho-take-html_10 (дата звернення 14.05.2020) – Назва з екрану.
- 14.Що таке CSS. – Електрон. дані (1 файл) – Режим доступу: https://developer.mozilla.org/uk/docs/Learn/CSS/First_steps/%D0%A9%D0%BE_%D1%82%D0%B0%D0%BA%D0%B5_CSS (дата звернення 14.05.2020) – Назва з екрану.
- 15.Cascading Style Sheets. – Електрон. дані (1 файл) – Режим доступу: <https://www.w3.org/Style/CSS/> (дата звернення 14.05.2020) – Назва з екрану.
- 16.Краткая история JavaScript. – Електрон. дані (1 файл) – Режим доступу: <https://habr.com/ru/company/livotyping/blog/324196/> (дата звернення 14.05.2020) – Назва з екрану.
- 17.What is JavaScript? A Basic Introduction to JS for Beginners. – Електрон. дані (1 файл) – Режим доступу: <https://www.hostinger.com/tutorials/what-is-javascript> (дата звернення 14.05.2020) – Назва з екрану.

18. Мова JavaScript та її можливості. – Електрон. дані (1 файл) – Режим доступу: <https://sites.google.com/site/webtehnologiitawebdizajn/mova-javascript-ta-ieie-mozlivosti> (дата звернення 14.05.2020) – Назва з екрану.
19. JavaScript. – Електрон. дані (1 файл) – Режим доступу: <https://astwellsoft.com/uk/blog/tehnology/javascript.html> (дата звернення 14.05.2020) – Назва з екрану.
20. What is Bootstrap: A Beginners Guide. – Електрон. дані (1 файл) – Режим доступу: <https://careerfoundry.com/en/blog/web-development/what-is-bootstrap-a-beginners-guide/> (дата звернення 14.05.2020) – Назва з екрану.
21. Bresenham's Line Generation Algorithm. – Електрон. дані (1 файл) – Режим доступу: <https://www.geeksforgeeks.org/bresenhams-line-generation-algorithm/> (дата звернення 15.05.2020) – Назва з екрану.
22. Bresenham's Line Algorithm. – Електрон. дані (1 файл) – Режим доступу: <https://www.javatpoint.com/computer-graphics-bresenhams-line-algorithm> (дата звернення 16.05.2020) – Назва з екрану.
23. DDA Line generation Algorithm in Computer Graphics. – Електрон. дані (1 файл) – Режим доступу: <https://www.geeksforgeeks.org/dda-line-generation-algorithm-computer-graphics/?ref=gp> (дата звернення 17.05.2020) – Назва з екрану.
24. DDA Algorithm. – Електрон. дані (1 файл) – Режим доступу: <https://www.javatpoint.com/computer-graphics-dda-algorithm> (дата звернення 17.05.2020) – Назва з екрану.
25. Anti-aliased Line | Xiaolin Wu's algorithm. – Електрон. дані (1 файл) – Режим доступу: <https://www.geeksforgeeks.org/anti-aliased-line-xiaolin-wus-algorithm/?ref=gp> (дата звернення 17.05.2020) – Назва з екрану.